Technical Report – Team 9

Speech Recognition System in Cars

TEAM 9	
NAME	**ASU ID**
Salil Batra	1207687659
Anant Srivastava	1209427008
Bhumika Mohan	1209374657
Chaoyi Fu	1209682471
Sagar Shivakumar	1209339829
Shujian Ke	1209577223
Siddharth Sivakumaran	1209381417
Monosrija Maity	1209381521

Table of Contents

Introduction

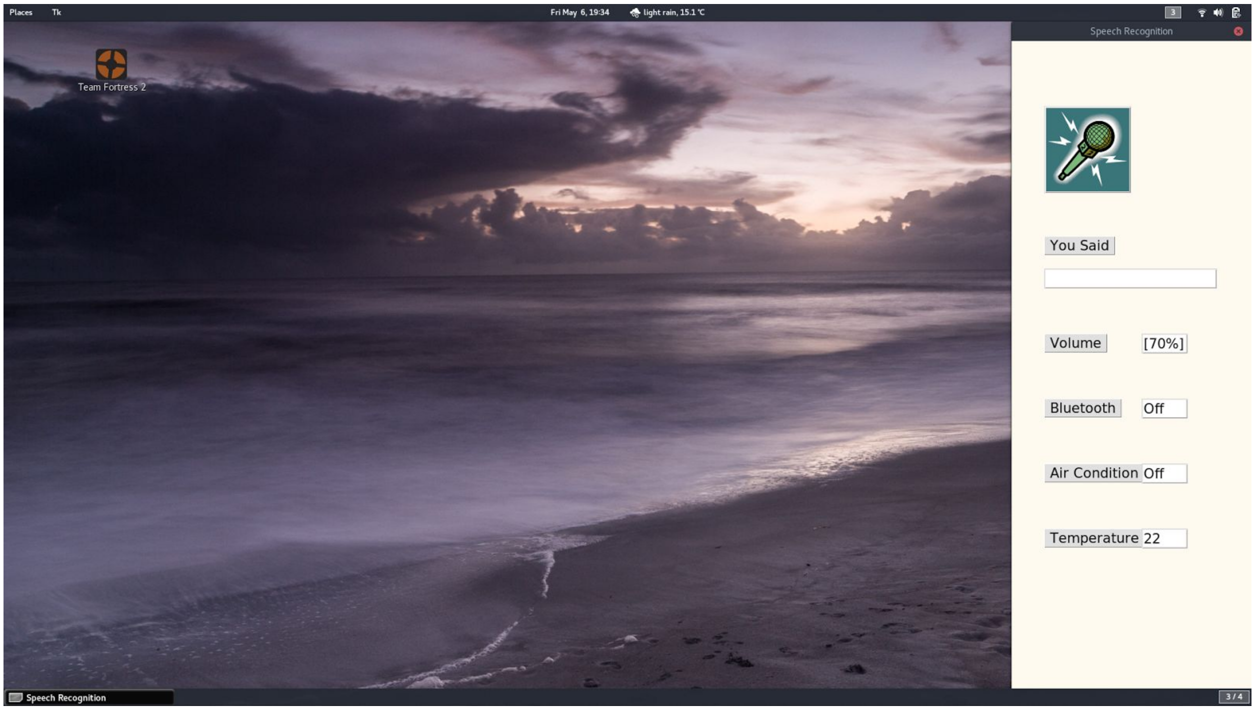
The software developed is a speech recognition software for infotainment system in cars. The software's primary functionality is to recognize a speech input and then perform the action according to the speech command. Our success criteria to develop such software was to focus on high accuracy of speech recognition and ease of use. We provide a user, functionalities such as switching on and off different car components, google search for a destination or nearest places of interest.

In this software we have also focused on providing a simple yet efficient GUI to the user so as to cause almost no distraction while driving the vehicle. In this technical report we will present details about the software design, its implementation, and a user guide on how to install the software and getting familiar with using the software.

Software Design Details

The focus of design in this software is to ensure negligible distraction to the driver.The software delivered is designed in Python using tkinter library. The GUI consists of a command input button to say a command, vertical layout of supported command (see next section for supported commands) inputs to check action status, and a horizontal display of the input command (when provided by the user). The input command button is a microphone picture which tells the user to click that button to input the command.

A sample GUI is shown in the image below:



Speech Recognition

You Said

Volume [70%]

Bluetooth Off

Air Condition Off

Temperature 22

Speech Commands

In this section we will discuss speech commands which are recognized by the delivered software.

Air Conditioning

To following commands provide support to perform actions on air conditioning.

- *"Switch AC ON"* :This command will switch on the air conditioning.
- *"Switch AC OFF"* : This command will switch off the air conditioning.

Vehicle Temperature

The following commands provide support to set the temperature inside the vehicle.

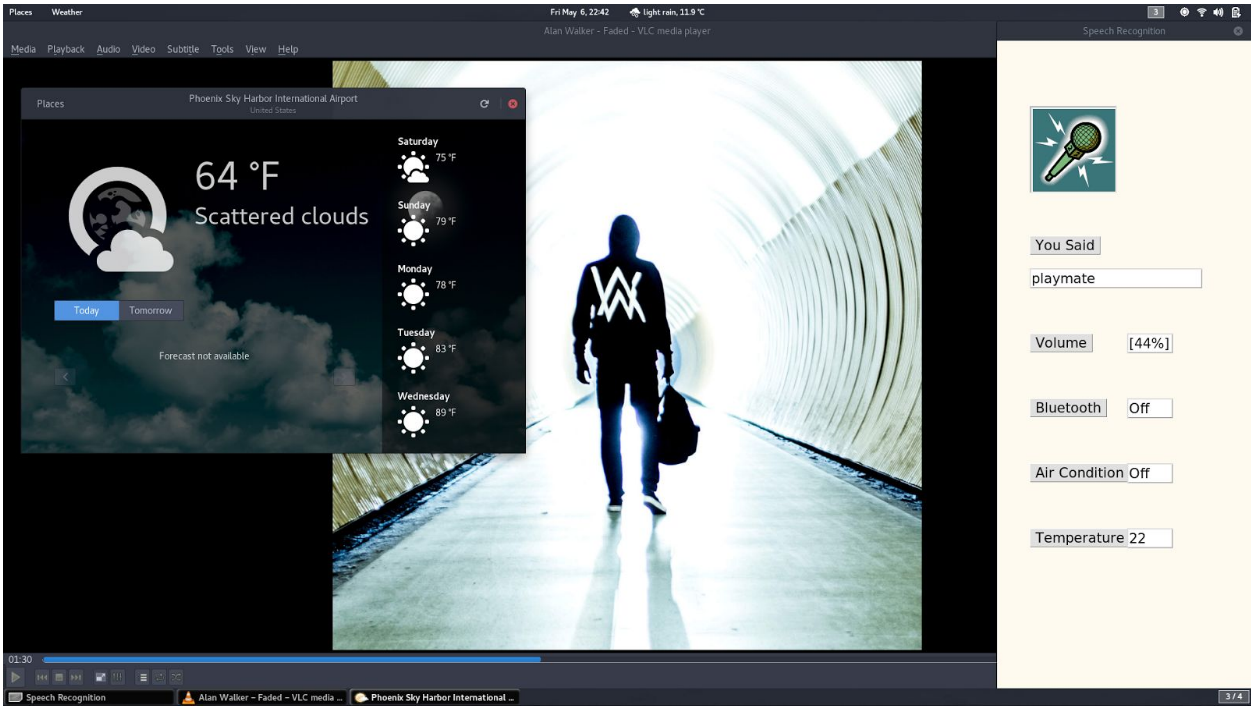
- *"Temperature UP"* : This command will increase the temperature.
- *"Temperature DOWN"* : This command will decrease the temperature.

Weather/Climate

The following command provide support to get information about the current temperature outside the car

- “*Show weather”:* It will display the current weather and climate conditions outside the car.

The image below shows the software displaying the weather based on the user’s location:



Speech Recognition

You Said: playmate

Volume: [44%]

Bluetooth: Off

Air Condition: Off

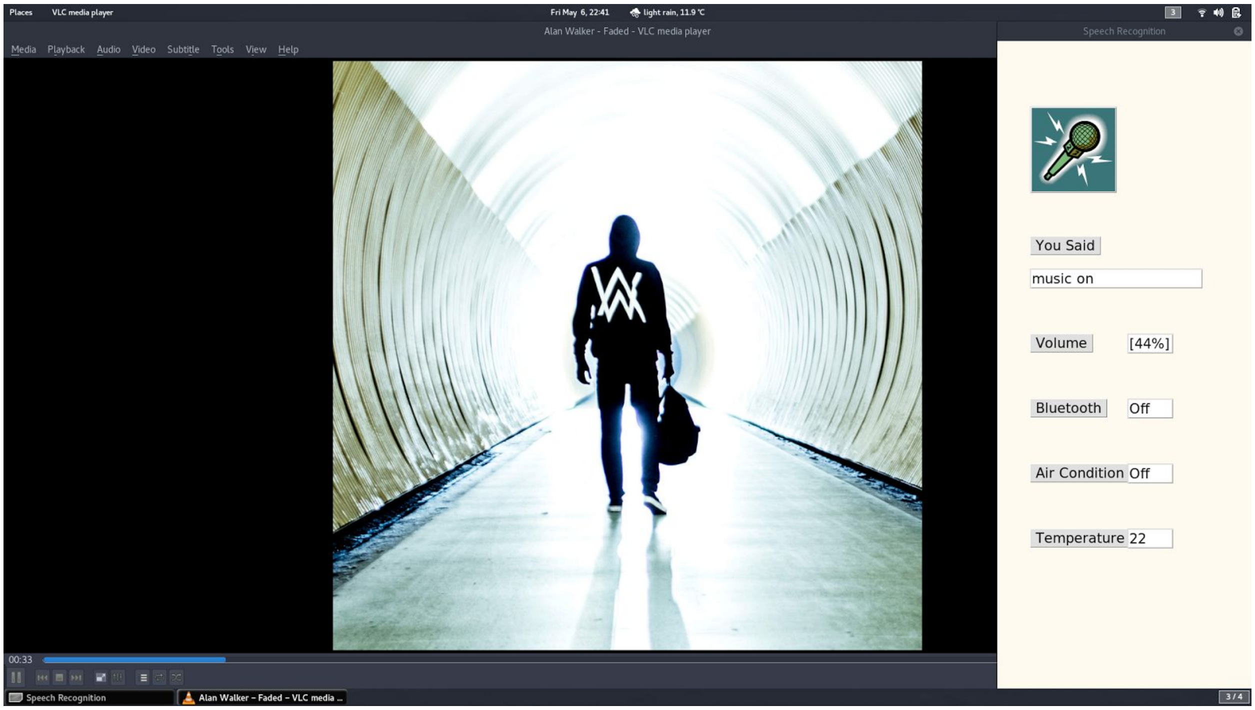
Temperature: 22

Music

The following commands provide support for playing music, increasing and decreasing the volume of the music player, and close the music player.

- *“Music ON”* : This command will switch the music player on.
- *“Music OFF”* : This command will switch the music player off.
- *“Volume INCREASE”* : This command will increase the volume of music player.
- *“Volume DECREASE”* : This command will decrease the volume of music player.

On executing the “Music On” command, the library selects the music present in the user’s playlist and starts playing them. The user can then increase or decrease the volume by using the commands specified above. The user can stop the music player by using the “Music Off” command. Below is an image showing the functioning of the music player.



You Said: music on

Volume: [44%]

Bluetooth: Off

Air Condition: Off

Temperature: 22

Bluetooth

The following commands provide support perform actions on bluetooth.

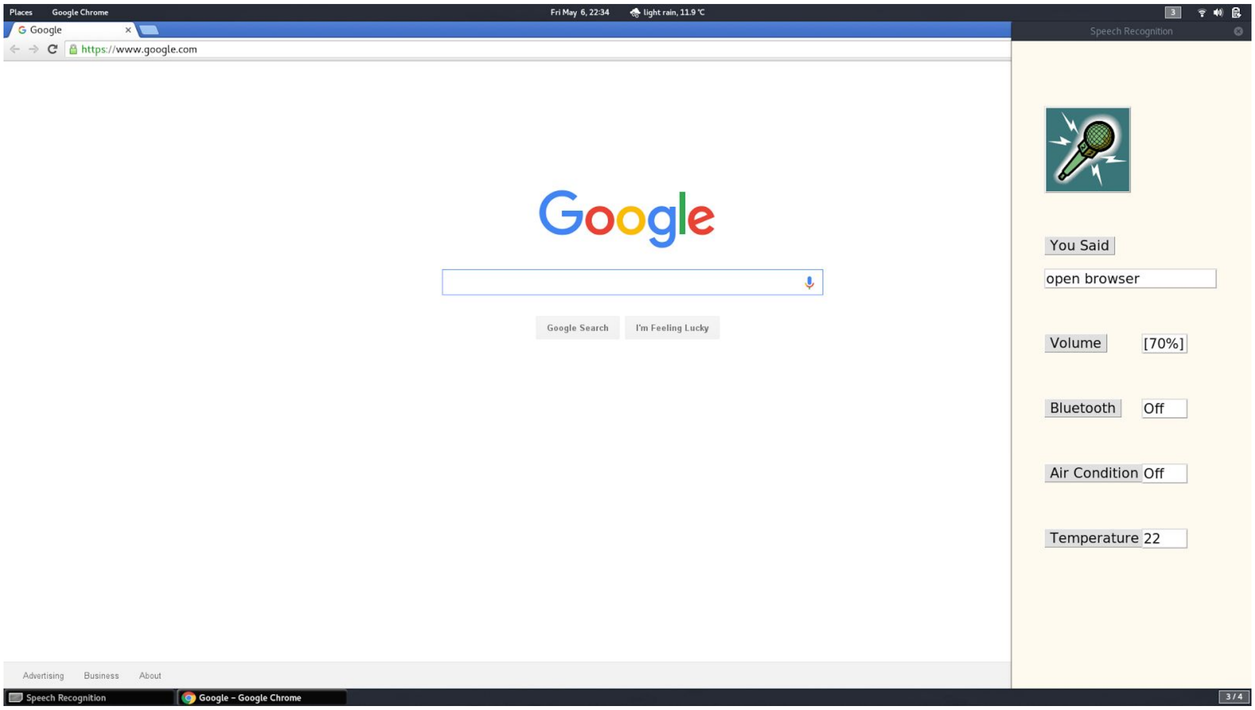
- *“Bluetooth ON”* : This command will switch the bluetooth on.
- *“Bluetooth OFF”* : This command will switch the bluetooth off.

Supported Commands to search on the internet.

- *“OPEN Browser” :* This command will open a browser for the user.
- *“Search <destination>”* : This command will take in the keyword search and the destination user was to drive to. The command will open the browser and will display source-destination map.
- *“Nearest <POI>”* : This command will take in the keyword search and the point of interest such as “*nearest coffee shop*”, “*nearest gas station* ”, etc., user was to drive to. The command will open the browser and will display the list of nearest POI spots along with selected source-destination map.

- *“Search <google>”* : This command will open the browser and display google search for the user’s input search command.
- *“CLOSE Browser”* : This command will close the browser.

The functioning of the ‘Open browser’ is shown in the image below:



You Said: open browser

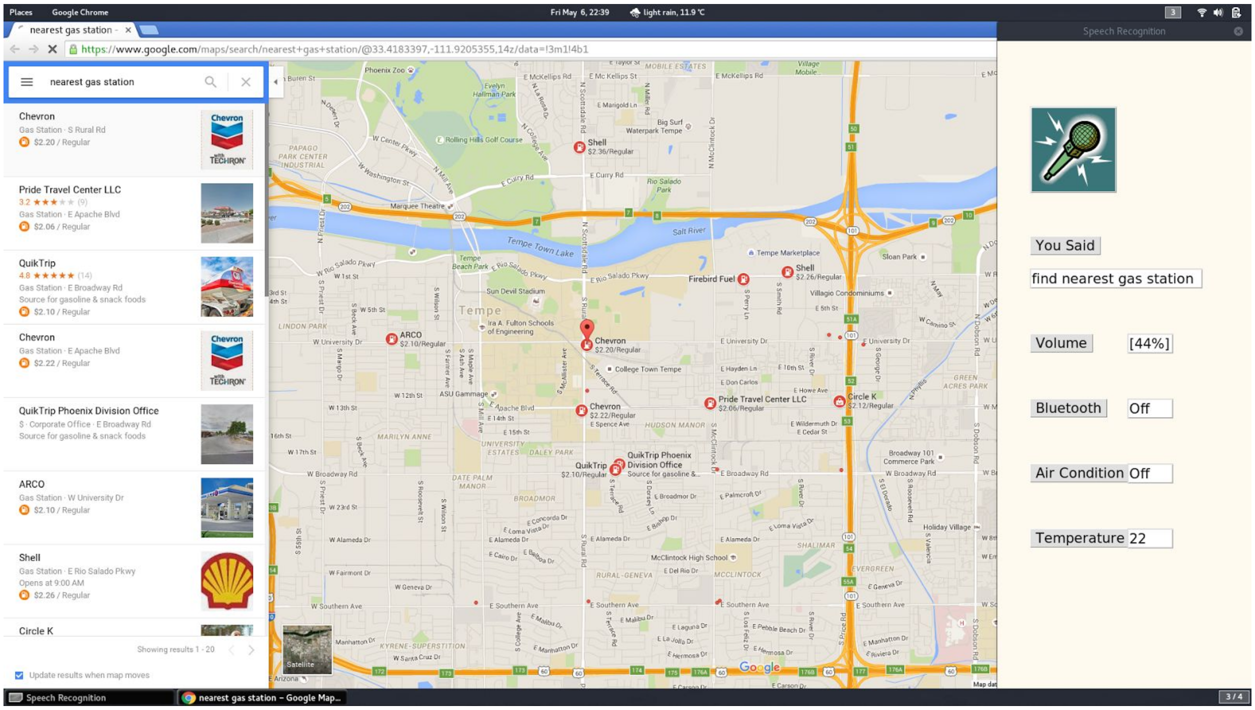
Volume: [70%]

Bluetooth: Off

Air Condition: Off

Temperature: 22

The next image uses the ‘find nearest’ command to search for the gas stations in the user’s vicinity by detecting the user’s location.



nearest gas station

https://www.google.com/maps/search/nearest+gas+station/@33.4183397,-111.9205355,14z/data=!3m1!4b1

Chevron
Gas Station · S Rural Rd
$2.20 / Regular

Pride Travel Center LLC
3.2 ★★★★★ (9)
Gas Station · E Apache Blvd
$2.06 / Regular

QuikTrip
4.8 ★★★★★ (14)
Gas Station · E Broadway Rd
Source for gasoline & snack foods
$2.10 / Regular

Chevron
Gas Station · E Apache Blvd
$2.22 / Regular

QuikTrip Phoenix Division Office
$ · Corporate Office · E Broadway Rd
Source for gasoline & snack foods

ARCO
Gas Station · W University Dr
$2.10 / Regular

Shell
Gas Station · E Rio Salado Pkwy
Opens at 9:00 AM
$2.26 / Regular

Circle K

Showing results 1 - 20

☑ Update results when map moves

You Said: find nearest gas station

Volume: [44%]

Bluetooth: Off

Air Condition: Off

Temperature: 22

Software Development Details

The software was developed in Python and Google API was used for speech recognition. The development focused on creating actions for the commands (described above), and achieve high accuracy for these commands. The software has different python scripts for GUI and backend (action for input commands). The minimum python version to run the software is Python3.x and the software is available on github. Every input command has been wired up to its respective function and a switch case module takes care of what function needs invoking. The software requires standard python dependencies and has been tested on Arch Linux operating system.

User Guide

This software provides high accuracy and ease of usability to the software. The software can be installed from the github repository. Minimum requirement to run the code is Python3.x and additional dependencies such as tkinter and google API if not already installed in python site-packages.

Once the software is compiled successfully, an interactive GUI will pop up and the user shall see a vertical layout of actionable commands.

To input a command, the user will have to click on the microphone button and speak the desired action to be performed. Every actionable command shows the current status and once the command is successfully executed, the status of - for example, air condition, will change state from on to off, or vice-versa.

To confirm the input command is correctly recognized, a horizontal text bar will display the recognized speech immediately. If the text is not what the user asked to perform, the user can click microphone button and provide the input again.

The software will open a browser every time a search input is given. It is to be noted that, once the browser is opened and the user wants to re-search on the browser, it is required to close the browser first and then input the command again.

Best Practices

- It is recommended that the user shall try to input short commands rather than a sentence. This is to achieve highly accurate results.
- Quiet surroundings ensures high accuracy of input command. Speech input while high outside noise, or high music volume can significantly decrease the software's efficiency.
- The software supports English language only.
- Mixing two commands will perform the action on first recognized command only. Input of individual commands is recommended.

Github repository : https://github.com/anantsrivastava30/speech_recognition.git