என் பேரு தமிழு (en̲ pēru tamiḻu): A Speech Recognition Robot for Tamil

Vasu Renganathan
University of Pennsylvania
vasur@sas.upenn.edu

0. Introduction

Speech recognition devices have been very popular in the recent years and they have reduced much of human's mechanical routines with simple and confined speech expressions. Amazon's Alexa, Google's 'Hi Google', Apple's Siri etc., are some of the popularly used tools that have been in the market for quite some time. Many of these devices recognize English speech considerably well within a limited structure with the basic idea that speech can be accounted for when it is within a pre-defined and a well-controlled construction. "Alexa! Set the alarm at 4:30PM", "Hi Google! What's the time now?", "Alexa! What is the weather now?", "Alexa! What is the status of the Flight AI177" etc., are some of the expressions which these gadgets can respond to in a very live manner. These devices when connected to online resources such as weather stations, atomic clocks, airline schedules and so on, they become very convenient in accessing information conveniently. When posed with some random expressions like, "Alexa! Why am I having a headache?", "Hi Google! Do I have COVID?", "Alexa! Why do I feel cold?" etc., they attempt to parse some of the key words such as "headache", "COVID", "COLD" respectively and come up with related answers to these key words as they can access from the web resources. These represent attempts to process any manageable structures from a mechanical perspective, without any human intelligence or rigorous Natural Language processing tasks involved. Thus, it can be assumed that leveraging the natural language utterances to the extent possible is one of the goals of this type of research and building such devices may not require much of AI techniques at all. Despite such simplicity in technology, this type of devices is not yet available for any Indian languages, including Tamil, although it is quite possible to build one by employing these technologies.

Along these lines of research, an attempt is made in this work to build a mobile Robot called தமிழு (tamiḻu), which can recognize a set of pre-recorded speech in Tamil and respond accordingly as coded in the algorithm written in C++ and Python. This is not precisely like any speech recognition devices as discussed above, which can parse words and syntactic structures of a language to certain extent. But, it is a pattern recognition tool which can match pre-recorded speech patterns to that of a similar one as uttered by the users. It is speaker centric in the sense that only the voice of those who recorded the speech can be recognized and responded, but not any random voice of such patterns. Its scope is very limited and mostly narrowed down to some of the scopes within a pedagogical context in that it can be used to train language learners to proceed through their carefully pre-recorded voice to casual spoken speech. However, a routine is provided in the system to replace the existing recordings with the voice of other users, so anyone can attempt to use this device as needed. Unlike the devices such as Alexa, Google, Sri etc., this is a mobile Robot which can be given command to move around with the four wheels and recognize objects on the way so it can divert its moving direction. This device is built with a speech recognition card and Arduino's technology. (Cf. https://www.arduino.cc).

என் பேரு தமிழு (en̲ pēru tamiḻu) – A Tamil Robot.

The Robot that is discussed here consists within it a C++ code interacting with the pre-recorded Tamil sentences, as stored in the speech recognition board, and the Arduino's resources to operate the motors as attached in the device; pinging any website to play audio as desired; responding to custom-tailored speech and so on. Although the potentials of this device can be expanded infinitely with other possibilities, particularly using the built-in Unix operating system with Wifi along with some of the potentials of the versatile software such as Python, PHP and the databases like MySQL, only a few built-in functionalities are discussed here. As this research is still in progress more features are being added upon on an ongoing basis to make it more robust for any imaginable practical applications. A set of template videos, mostly developed toward language learning purposes, is demonstrated at http://robot.tamilnlp.com and subsequently described in detail in this article.

0.1. Moving around with L298N Motor Drive Controller and Tamil Commands:

The L298 motor drive controller as attached to Arduino boards can be used to make mobile robots and this technology is made use of in the Tamil Robot with a driving factor being a set of Tamil commands interacted through the speech recognition module. The commands such as "**முன்னால வாங்க**" (mun̲n̲āla vāṅka)[1], "**பின்னால போங்க**" (pin̲n̲āla pōṅka), "**சுத்துங்க**" (cuttuṅka), "**எடது பக்கமா சுத்துங்க**" (eṭatu pakkamā cuttuṅka), "**வலது பக்கமா சுத்துங்க**" (valatu pakkamaa cuttuṅka) etc., are some of such pre-recorded speech patterns in the recognition board and they can be used to operate the Robot to move in different directions such as forward, backward, left, and right. When the board's speech recognition microphone coupled with the C++ code's loop structure, a sequence of commands can be given to the Robot to move around accordingly. This feature can be used conveniently by the learners of Tamil to get used to directions in Tamil and subsequently be able to operate the Robot to move around with their own voice. As mentioned earlier, this Robot is built in such a way that anyone can replace the existing commands with their own voice and subsequently make the Robot follow their own directions. A module to give directions and getting used to corresponding direction related expressions in Tamil is demonstrated in the video http://robot.tamilnlp.com/directions.mp4.

0.2. Repeat until you learn: Learning Tamil literature with the Tamil Robot.

With the use of the in-built functions to play sound in Arduino and correspondingly with the use of Python's sound libraries, it becomes possible to play pre-recorded speech efficiently in a number of different ways. 'Repeating the lines of verses until one learns them fluently' is a method that is employed to train students learn Tamil literary verses such as **ஆத்திசூடி** (āṭṭicūṭi) and **புறநானூறு** (puṟanāṉūṟu) to certain extent. Some of these poems are recorded with each line of the verses as separate recognition unit, and subsequently the students are capable of repeating

[1] Polite expressions are used to address this Robot for two reasons. One, for using a gender-neutral expression and for the other to make unambiguous utterances to make the recognition easier for the Robot. In fact, nothing prevents the user from replacing them with their own expressions of choice, using the corresponding routine but by taking a careful measure to avoid any ambiguity for the pattern matching process by the Robot.

them by one line at a time, as played by the Robot. When the student utters a line that is recognizable by the Robot, it plays the subsequent line for the student to repeat, so the student can repeat the same line, or the line next to it. Thus, when the student utters the following line, instead of repeating the same line, the Robot continues to utter subsequent lines successively. Thus, the Robot and the student get to recite the entire poem with each of them saying a line at a time, a stage which can be conveniently called a "fully memorized stage" for students. This method of "Repeat until you learn" is demonstrated in the videos http://robot.tamilnlp.com/aattucudi.mp4 and http://robot.tamilnlp.com/sangam.mp4, with lines from Atticudi and Purananuru respectively. This method can be made use of to memorize poems from any Tamil literary piece, provided the lines of the poems are recorded in the speech recognition board ahead of time and made available for the Robot to recognize. This method can also be used to train students learn and memorize words such as Tamil months, days, Tamil years and related others, which require a constant practice. For instance, the video http://robot.tamilnlp.com/tamil_months.mp4 demonstrates as to how Tamil months are practiced with a student either repeating after what is said by the Robot or alternate with Robot to utter them all.

0.3. Live connection with online database servers.

Not many Arduno's boards come with built-in Wifi and operating systems, except for the board Yun (https://www.arduino.cc/en/Guide/ArduinoYun). What is unique about Yun is that it has a built-in full-blown Linux operating system along with all the Wifi capabilities. With Linux operating system, it is virtually possible to make use of all the functionalities of any mini-computer including to run software applications such as Python, PHP, Shell Script and so on; and interact with databases as needed. Raspberry PI also has similar capabilities, and it is quite possible to use the speech recognition card with Raspberry PI, but it hasn't been attempted yet in this work. Not to mention the fact that a wide variety of audio and video libraries can also be made use of with the Arduino and Speech recognition boards provided they are linked with the Yun board. Exploiting these functionalities of Yun, this Robot interacts with an online database dynamically and play audio files from English-Tamil pedagogical dictionary and Thirukkural verses stored in audio format. With the commands such as, "தமிழு திருக்குறள் சொல்லுங்க" (tamiḻu tirukkuṟaḷ colluṅka), "அகராதிச் சொல் சொல்லுங்க" (akarātic col colluṅka), the Robot picks verses from Thirukkural and words from dictionary randomly and plays it for the user. Although this system is not furnished with many other internet based features such as querying weather, flight schedules etc., as Alexa and other devices do, it is quite possible to enable such features in this system by using the corresponding APIs. However, with an ongoing update of the online databases for Tamil, it is quite possible to make the Robot serve new content dynamically without having to make any change in the Robot itself. What it entails is that Robot on the one hand, voice commands and interaction with the database servers on the other hand make a live interaction with the knowledge content as stored in the databases.

0.4. Interactive conversational partner.

Conversations between Robot and the user can be made possible provided a set of dialogues are written ahead of time and subsequently one part of the dialogue is pre-recorded in the recognition card and the other part is communicated by the user. Every dialogue consists of human's turn versus Robot's turn and the human's turn should be recognizable by the Robot as

pre-recorded in the recognition board. Hence, all the human's turns when recognized, the Robot plays the corresponding response in a real-time mode. Such conversations need not necessarily be performed in a sequence from beginning to end and the Robot can be made to respond to any part of the conversation at any given time. A sample interactive conversation dialogue is made use of with the Tamil Robot and demonstrated in the video: http://robot.tamilnlp.com/beginners_tamil.mp4. Given the fact that there are complex expressions in Tamil which are dificult to learn, a separate dialogue is written to particularly practice a number of patterns involving many complex grammatical constructions in Tamil. For example, expressions like பார்த்துக்கிட்டிருக்கிறதற்காகவாவது (pārttukkiṭṭirukkiṟataṟkākavāvatu), வறேண்ணுட்டாங்களா? (vaṟēṇṇuṭṭāṅkaḷā?), போயிடமுடியாதுங்றதுக்காகவா (pōyiṭamuṭiyātuṅṟatukkākavā) are some of the commonly used complex expressions by native speakers of Tamil in daily routines but are very hard to learn and utter by any second language learner. When such expressions are recorded and kept in the recognition board, the Robot may be made to recognize such expressions from the student and respond accordingly, in the same technique of 'repeat until you learn'. Robot repeats the same expression until it is uttered correctly by the student and does not proceed to subsequent expression unless it is pronounced correct. A sample video along this line of method is demonstrated in http://robot.tamilnlp.com/learn_to_speak.mp4.

0.5. Talking Robots

With a sequence of commands and responses being the driving factor in constructing these Robots, it is surely possible to let the Robots interact each other provided the question/answer sessions are planned carefully with appropriate time intervals. An attempt is made to make two Tamil Robots interact each other with simple conversation and it is demonstrated in the video http://robot.tamilnlp.com/tamil_sangam.mp4. Although it is quite possible to make a long conversation between Robots along these lines, such effort has not been attempted yet. The following four key steps would help understanding the process involved in the performance of this Robot.

1) Robot Listening Mode → Audio Signal ← Users Issuing Commands

2) Pattern Matching with Pre-Recorded Dynamically Replaceable Tamil Commands
3) Perform an action with a match
(or)
4) Go to the listening mode with a signal for failure.

Step 3 can be one of the actions as outlined in sections 1.1 through 1.5, and the listening mode and issuing commands as in step 1 should be concurrent and happen simultaneously. It is to be noted that while performing the action as in step 3, the listening mode and subsequently issuing commands would be stopped, unless it is done in a loop so issuing commands can be done while performing an action by the Robot. Thus, the part of issuing commands can be performed by a human or by another Robot, provided the commands are issued carefully synched with the listening mode.

0.5.1. Commands and Responses:

The recognition board consists of twelve units and each unit can hold within it thirty unique commands, with a total of three hundred and sixty commands. The commands as stored in each unit should be unique in nature and should not be ambiguous in any manner for a good result. Special care needs to be taken to make the commands without any ambiguities involving identical words and expression. Each time a recognition is made within a single unit with what is uttered corresponding action is performed by the robot. Separate command needs to be included in each unit to switch to other units to perform actions within them. For example, the first unit has the trigger word **வணக்கம்** 'vaṇakkam'. Upon recognizing this command, the other commands as stored within the first unit can be accessed. Subsequently to switch between other units, appropriate command needs to be included in each unit. To cite an example, when the command **ஆத்திசூடி சொல்லுங்க** 'ātticūṭi colluṅka', is recognized, the unit with the commands for **ஆத்திசூடி** 'ātticūṭi' will be called upon and subsequently all the actions related to it can be performed within it. Similarly, when the command **புறநானூறு சொல்லுங்க** 'puṟanāṉūṟu colluṅka' is recognized, it will be switched to the unit where **புறநானூறு** 'puṟanāṉūṟu' verses are stored and correspondingly all responses related to it can be engaged in. When commands like **பாடுங்க** 'pāṭuṅka', **எம்ஜியார் பாட்டு பாடுங்க** 'emjiyār pāṭṭu pāṭuṅka' etc., are recognized, corresponding audio files are played using Python's audio library. Thus, recognition and corresponding actions are compartmentalized within the commands as stored within twelve units and interactions between human and robot within each of these units should be well planned ahead of time as to how they are accessed. In this sense, the Robot is made to act upon within a set of twelve domains of speech situations, and switching between these domains need to be determined ahead of time. Perhaps, it might be possible to write a machine learning algorithm to switch between units by the Robot itself based on the users multiple responses, but that sort of attempts have not been made yet in this system.

0.6. Conclusion

It is quite possible to let a robot understand and process commands of any complex type but doing so would require a very sophisticated and complex natural language processing technique built as part of the code. Particularly, such attempts should involve within it many NLP processors such as text to speech, speech to text, morphological and syntactic analyzer and so on, but such attempt is yet to be made in this work. However, the intricacies and processes involved in building such NLP methods have been studied and demonstrated already on various other contexts including text to speech, morphological and syntactic analyzer etc., as can be seen in the works of Renganathan (2016, 2014, 2001). What is demonstrated in this work is a unique pattern matching method constructed within a set of Tamil speech patterns. All possible scopes of this work are demonstrated with a set of sample videos as filmed in a live mode with the Tamil Robot called **தமிழு** (tamiḻu).

1. References[2]

Similar technological systems consulted:
Amazon's Alexa - https://en.wikipedia.org/wiki/Amazon_Alexa.
Google Assistant - https://assistant.google.com.
Apple's Siri - https://www.apple.com/siri/

Renganathan, Vasu (2016). Computational Approaches to Tamil Linguistics, Cre-A. Chennai, India.
Renganathan, Vasu (2016). Computational Approaches to Tamil Linguistics: Scopes and Prospects. In the proceedings of the 15th Tamil Internet Conference, Gandhigram Rural University, Dindigul, Tamil Nadu. (http://www.uttamam.org/papers/16_02.pdf).
Renganathan, Vasu (2014). Computational Phonology and the Development of Text-to-Speech Application for Tamil. In the Proceedings of the International conference on Tamil Internet, 2014, Pondicherry, India. (http://www.uttamam.org/papers/14_35.pdf).
Renganathan, Vasu (2001). Development of Morphological Tagger for Tamil, In the Proceedings of the International Conference on Tamil Internet 2001, Kuala Lumpur, Malaysia. (http://www.uttamam.org/papers/01_34.pdf).

[2] The online resources as referred to in this article were accessible on 11/24/2021.