

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Intelligent Voice Assistant Using Android Platform

Sutar Shekhar S¹

Information Technology Department
Sinhgad Academy of Engineering
Pune – India

Kamad Neha S³

Information Technology Department
Sinhgad Academy of Engineering
Pune – India

Pophali Sameer S²

Information Technology Department
Sinhgad Academy of Engineering
Pune – India

Deokate laxman J⁴

Information Technology Department
Sinhgad Academy of Engineering
Pune – India

Abstract: The software development on Smartphone is very promising. The common way of communication used by people in daily life is through the speech. If the mobile phone can listen to the user for the request or handle the daily affairs, then give the right response, it will be much easier for users to communicate with their phone, and the mobile phone will be much more “Smart” as a human assistant.

This project is focusing on the Android development over the voice control (recognition, generate and analyze corresponding commands, intelligent responses automatically), Google products and relevant APIs (Google weather, Google search and etc), Wikipedia API and mobile device references ranging from Speech-To-Text, Text-To-Speech technology, camera, messaging and other services and technology that are needed in daily life. Physically disable person can also use this Application. Voice SMS is an application developed in this work that allows a user to record and convert spoken messages into SMS text message. Speech recognition for Voice uses a technique based on hidden Markov models (HMM - Hidden Markov Model). It is currently the most successful and most flexible approach to speech recognition.

Keywords: HMM (Hidden Markov Model), TTS (Text to Speech), Intent, Android OS.

I. INTRODUCTION

Many years ago, software programs were developed and run on the computer. Nowadays, smart phones are widely used by all people. About 35 percent of the Americans have some sort of Smartphone. This shows that the market is increasing fast and there are also more capabilities for Smartphone because of this wide use. This project is originated from a popular application from Apple called “Siri” and Google voice Action. Siri application was released on the date when the iPhone4S was published. Google Voice Action can be used in any android version above 4.4.

In this work we have developed an application for sending SMS messages, opening inbuilt Applications like gallery, settings, camera, messaging and etc which uses Google's speech recognition engine. The main goal of application is to use user voice commands to access smart phone instead of using it manually. As it integrates most of the mobile phone services for daily use, it could be useful for getting a more convenient life and it will be helpful for those people who have disabilities for manual operations. This is also part of the reason why it has been chosen as the degree project.

In this application user is able to access the services of smart phone with their voice command. User can easily send a message to the recipient available in their contact list as well as to the mobile number by voice command. In this Project physically disable person or the person having less knowledge about smart phone or how to access the smart phone can easily access the phone with their voice or speech command.

II. ANDROID ARCHITECTURE

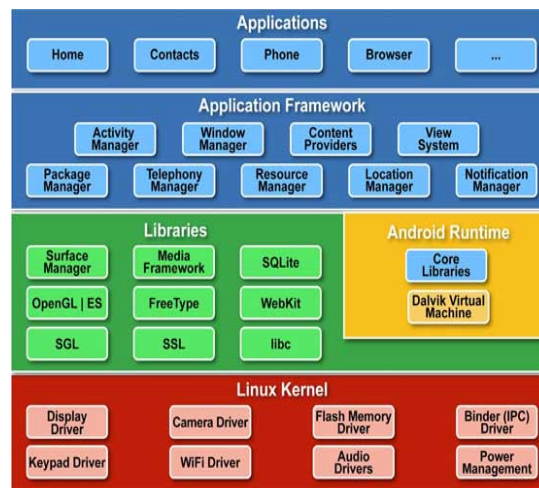


Fig. 1 Architecture of Android platform

Android operating system is divided into five sections and four main layers as shown in the above diagram. At the bottom layer Linux Kernel (Linux 2.6) is present with approximately 115 patches which provide basic system functionality like process management, memory management, device management which is used to manage input & output devices. It also handles networking vast array of device drivers, which makes it easy for interfacing to peripheral hardware.

Next to Linux kernel there is library set which contains SQLite database which is a useful repository for storage and sharing of application data, open-source Web browser engine WebKit, well known library libc, libraries to play and record audio and video, SSL libraries responsible for Internet security etc.

Next section is Android Runtime available on the second layer from the bottom. This section gives a key component called **Dalvik Virtual Machine** which is like a Java Virtual Machine designed and optimized for Android. By using Linux core features like memory management & multi-threading The Dalvik VM enables every Android application to run in its own process, with its own instance of the Dalvik virtual machine. It also provides a set of core libraries due to which one can write android application using standard java programming language.

Next layer is Application Framework layer gives many higher-level services to applications in the form of Java classes. Developers are allowed to make use of these services in their applications.

Next layer is Application layer. One can find all the Android application in this layer. One can write the application to be installed on this layer only. Examples of such applications are Contacts Books, Browser, and Games etc.

III. SYSTEM SCOPE

This paper proposes the IND-OCPA-P model to analyze the security of the proposed EOB and the encryption schemes supporting an efficient range query over encrypted data.

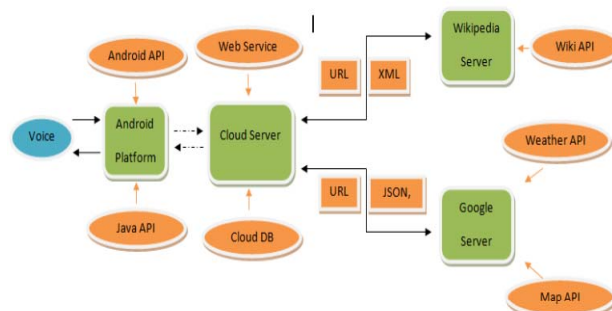


Fig. 2 Technologies and technique used while developing application

The developing simulation is shown in Fig. 2 architecture. The architecture also depicts the technologies used and technique used in the project.

- » Firstly, the acoustic voice input will be recorded by the Android phone.
- » The voice input will be processed by the application through recognizer Intent which is present in Android API and JAVA API. A recorded plain text will be generated and will be sent to the cloud server or Android application itself depending on the command given by user.
- » If it is a search request then plaintext command will be forwarded to the cloud server which decodes the received text with the Java API, references, and predefined database, then decide the following procedures that should be executed.
- » A command will be generated into a URL by the cloud server and sent to the specific server (Google server, Wikipedia server).
- » The server which receives the request will use the specific API of Wikipedia API, Google API to generate the response in XML or JSON format.
- » Cloud server will obtain the XML/ JSON response file and transform to a specific response which will be led to the Android application.
- » The Android application will generate the audio output to the users with the mobile speaker.

IV. SYSTEM FUNCTION

This application includes the functions and services such as: calling services, text message transformation, mail exchange, alarm, event handler, location services, music player service, checking weather, Google search, Wikipedia search, camera, Bluetooth headset support and help menu. The list below indicates the information and the requirements of each individual function.

The program has two modes to well fetch the services and functions. The program will start with voice mode as its primary mode to provide the voice assistant, but the user can select switching to the text mode if he or she is not well working with the voice mode or the surrounds don't support the voice recognition well i.e. in noisy atmosphere.

- » Text Message transformation, users are able to send the SMS to a specific person in the phonebook or to the new contact as well by giving a correct command which contains the messaging request keywords along with the destination person details, the message should be sent to the destination immediately
- » Calling service, the application should allow the users to make a call to the person in the contacts or by saying mobile number of the person to whom user wants to call. By giving a correct command with the calling request to a stored person, the Android phone should successfully direct to the number of the person requested
- » Alarm, as a basic function on the mobile phone, it is frequently that users need to set the alarm to a specific time. The user could set the alarm through the request with the given time.
- » Mail exchange, users are able to send the mail to the person with mail address in the contacts. By giving a correct command containing the mail request keyword along with email id of the destination person; the mail should be received by the recipient after it has been sent.
- » Google search, the search will enable the user to search anything on Google. The search will give result list back and displayed on the browser.
- » Camera, the camera function will invoke the camera on the mobile phone to take a picture of the current view, the picture will be stored in the Gallery for later viewing and operation.

V. SPEECH RECOGNITION

Speech Recognition for intelligent voice assistant application is done using Google Server, using HMM algorithm [1]. This process involves the conversion of acoustic speech into a set of words and is performed by software component. Accuracy of speech recognition systems differ in vocabulary size and confusability, speaker dependence vs. independence, modality of speech (isolated, discontinuous, or continuous speech, read or spontaneous speech), task and language constraints. The system consists of five modules: feature extraction, HMM phone model training, dictionary preparation, grammar estimation, and sentence decoding [1].

Speech recognition systems which are based on Hidden Markov Models which are today's most widely applied in modern technologies. They use the word or phoneme as a unit for modeling. The model output is hidden probabilistic functions of state and can't be deterministically specified. State sequence through model is not exactly known.

VI. MAIN PARTS OF SYSTEM

1) *Voice Recognition:*

To recognise a speech input we have to give permission to the application i.e. RECORD_AUDIO in Android Manifest file. Voice_Recognition Activity is startup activity defined as main and launcher in AndroidManifest.xml file. REQUEST_CODE is static integer variable, declared on the beginning of activity and used to confirm response when engine for speech recognition is started. REQUEST_CODE has positive value. Results of recognition are saved in variable declared as ListView type.

Method onCreate is called when activity is initiated. Most of the initialization goes in this method. We have to set layout for our activity in this method. This is accomplished with command setContentView(R.layout.voice_recognition).

If a mobile device doesn't have one of many Google's applications which integrate speech recognition, further work of this application Voice SMS will be disabled and message on the screen will be "Recognizer not present". Recognition process is done through one of Google's speech recognition applications. If recognition activity is present user can start the speech recognition by pressing on the mic image and thus launching startActivityForResult (Intent intent, int requestCode). The application uses startActivityForResult() to broadcast an intent that requests voice recognition, including an extra parameter that specifies one of two language models. Intent is defined with intent.putExtra (RecognizerIntent).

EXTRA_LANGUAGE_MODEL is set as US-EN by default.

2) *Messaging:*

The Messaging class uses the permissions READ_MESSAGE and WRITE_MESSAGE. In this application will ask the user that "Do you want to send message to existing contact or to new number?". After recognizing user response the speech to text conversion takes place and phonebook will be scanned for the contact number. If user wants to send message to new number, he can speech whole number. Upon confirmation of the recipient the application will ask for the content of the message. User can speak whatever message he wants to send, then upon conforming correctness of the message it will be sent to the desired recipient.

When we receive any message Broadcast Receiver will listen to the event. Then application will read out entire message for you.

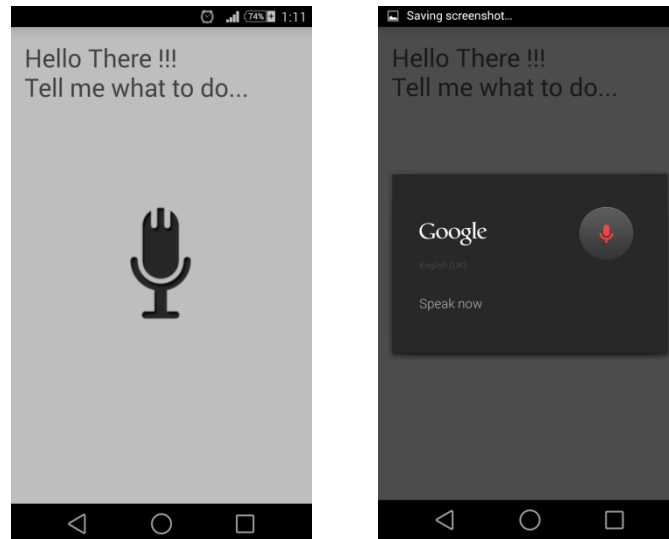


Fig. 3 Starting of Voice Recognition Activity after pressing mic present on the application

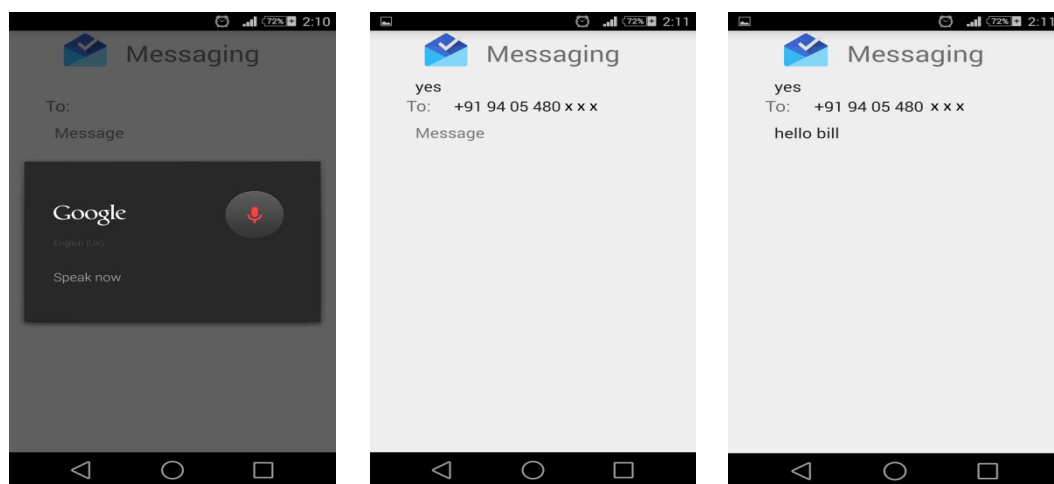


Fig. 4 Messaging functionality in the application

VII. APPLICATION FUNCTIONALITY

In this application user can use all the inbuilt applications in the android phone using their voice command. Voice or Speech will be given as input to the application. When application starts up it will show a button which will start the voice recognition process. When speech has been detected application opens connection with Google API.

By this user can easily use the inbuilt facility like message, when the user speak message then the application will ask whether the recipient is form existing contact say yes or no. If it is from existing contact then it will ask the recipient name and after that application will ask whether check the recipient is correct say yes or no. If yes then speak the message u want to want to send and the voice message will be converted in the form of plain text[2][3] that is speech to text, application will ask the confirmation about sending the message if it is yes than the message will be send to the corresponding recipient. As shown in the fig [4]. Application will also read the incoming message that is it will converted to speech[4][5].

Application is also able to open all the inbuilt application like gallery, settings, calculator, alarm, and camera. Even user can open other options like browsing. The inbuilt application will be open by just speaking the application name by voice command respectively.

VIII. CONCLUSION

We have develop application, in which user can easily send a message with their voice command and also tried to use most of the inbuilt application with voice command. But all these application have adaptation for the English language. we tried to

develop application in such a way that it will be useful to the visually impaired persons as the whole of our system work with a voice command. We are also surveying to use mailing and calendar where user will be able to mail and also create a event with the voice command.

The Objective for future development is to develop a voice engine which can also work for the other local language like Hindi or Marathi.

ACKNOWLEDGEMENT

We are highly indebted to Mr. Pramesh Patil (Persistent Systems Pvt. Ltd.) for his guidance and constant supervision as well as for providing necessary information.

We would like to express our gratitude towards Dr. V. M. Wadhai and Prof. A. N. Adapanawar for their kind co-operation and encouragement.

References

1. Yunxin Zhao, A Speaker-Independent Continuous Speech Recognition System Using Continuous Mixture Gaussian. IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING, VOL. 1, NO. 3, JULY 1993.
2. Brandon Ballinger, Cyril Allauzen, Alexander Gruenstein, Johan Schalkwyk, On-Demand Language Model Interpolation for Mobile Speech Input INTERSPEECH 2010, 26-30 September 2010, Makuhari, Chiba, Japan, pp 1812-1815.
3. Ryuichi Nisimura, Jumpei Miyake, Hideki Kawahara and Toshio Irino, —Speech-To-Text Input Method For Web System Using JavaScript_, IEEE SLT 2008 pp 209-212.
4. S. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. In IEEE Transactions on Acoustics, Speech and Signal Processing, volume 35, pages 400-401, March 1987.
5. D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah. Boosted MMI for model and feature-space discriminative training. In Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), 2008.
6. <http://developer.android.com/index.html>
7. www.stackoverflow.com
8. <http://www.tutorialspoint.com/html/>

AUTHOR(S) PROFILE



Sutar Shekhar Subhash, Sinhgad Academy of Engineering, Pune.



Pophali Sameer Shekhar, Sinhgad Academy of Engineering, Pune.



Kamad Neha S., Sinhgad Academy of Engineering, Pune.



Prof. Deokate Laxman J., Sinhgad Academy of Engineering, Pune.