

Implementation of Android Based Speech Recognition for Indonesian Geography Dictionary

Hugeng Hugeng¹, Edbert Hansel²

Department of Computer Engineering, Universitas Multimedia Nusantara, Tangerang, Indonesia

¹hugeng@umn.ac.id, ²edbert.hansel@gmail.com

Diterima 7 Desember 2015

Disetujui 28 Desember 2015

Abstract—We have built an application of speech recognition for Indonesian geography dictionary based on Android operating system, named GAIA. This application uses a smartphone as a device to receive input in the form of a spoken word from a user. The approach used in recognition is Hidden Markov Model which is contained in the Pocketsphinx library. The phonemes used are Indonesian phonemes' rule. The advantage of this application is that it can be used without internet access. In the application testing, word detection is done with four conditions to determine the level of accuracy. The four conditions are near silent, near noisy, far silent, and far noisy. From the testing and analysis conducted, it can be concluded that GAIA application can be built as a speech recognition application on Android for Indonesian geography dictionary; with the results in the near silent condition accuracy of word recognition reaches an average of 52.87%, in the near noisy reaches an average of 14.5%, in the far silent condition reaches an average of 23.2%, and in the far noisy condition reaches an average of 2.8%.

Index Terms—*speech recognition, Indonesian geography dictionary, Hidden Markov Model, Pocketsphinx, Android.*

I. Introduction

From the Regulation of Indonesia Minister of National Education number 22 year 2006 about content standards for basic secondary education

units, geography is a branch of science that examines on earth, aspect, and the processes that shape it, causal and spatial human interaction with the environment, and human interaction with the place. By studying geography, intellectual ability of each person who studies it, can be developed. Geography can increase the curiosity, the power to observe the natural environment, to train memory and imagery to life with its environment, and can train the ability to solve the problems of life that occur daily or explicitly geography has a high educational value. Through a geography lesson, the cognitive, affective, and psychomotor ability can be improved so that everyone who learns it can reach the mental maturity in thinking, feeling, and developing skill.

Previous research in [1] produced an Android application for Indonesian geography dictionary. However, the application does not include speech recognition feature. The other research [2] produced an Android speech recognition application for Indonesian language. Unfortunately, the application is limited only for speech to text and uses English phonemes' rules.

Based on the previous researches above, we create a speech recognition application for Indonesian geography dictionary named "GAIA". The name of GAIA is taken from the name of the earth goddess in Greek mythology; GAIA was the great mother of all or creator and giver of birth to the earth and all the universe. GAIA application is built on a smartphone with Android operating system. The phonemes used are Indonesian

phonemes' rule as used in [3]. The advantages of this application is that it can be used without internet access and includes a speech recognition feature so it is expected to make student more interested in learning geography in Indonesian terms. Through this research, GAIA application is expected to be useful for Indonesian students to learn and to gain knowledge about the terms of geography.

II. Literature Review

A. Android Operating System

Android is a Linux-based operating system designed specifically for touch screen mobile communication devices such as smartphones and tablets. Android was developed by Google in collaboration with Open Handset Alliance. Android supports open source projects so that many Android users can build their own applications on Android phones or tablets. Android developers provide the Android Software Development Kit (SDK) and Native Development Kit (NDK) which can be obtained free of charge for any developer who wants to develop Android-based applications [4].

B. Speech Recognition

Speech recognition is the development of techniques and systems that enable a computer to accept input in the form of spoken words [5]. This technology allows a device to recognize and understand the words that are spoken by digitizing words and match the digital signal with a certain pattern stored in the device. The spoken words are transformed into a digital signal by changing the sound waves into a set of numbers which are then adapted to the specific code to identify the words. The results of the identification of the spoken words can be displayed in the form of text or it can be read by the device technology as a command to do a job, for example, pressing a key on the handset which is done automatically by voice command.

C. Pocketsphinx

Pocketsphinx is a mobile version of speech recognition application library of the Sphinx system designed by Carnegie Mellon University (CMU) [6]. In this library, the approach used in the speech recognition system is Hidden Markov Model (HMM). The process of learning sound units is called training, while the process of using the gained knowledge to deduce the most probable sequence of units in a given signal is called decoding, or simply called recognition. Because there are two processes then SPHINX trainer and SPHINX decoder are required.

Phonetic writing on Pocketsphinx using Arpabet system. Arpabet is a phonetic transcription that was developed by the Advanced Research Projects Agency (ARPA) as part of the Speech Understanding Project. Arpabet represents each phoneme of the General American English with a different sequence of ASCII characters. In Arpabet, each phoneme is represented by one or two letters.

D. Hidden Markov Model

Hidden Markov Model (HMM) is a statistical model of a system that is assumed to be a Markov process with unknown parameters or states. Hidden states must be determined from the states that can be observed. Then, the specified states can be used for further analysis, for example, for pattern recognition. In HMM, a state cannot be observed directly, but one can observe variables that are affected by the state. Each state has a probability distribution over the output tokens that may arise. Therefore, a series of tokens generated by the HMM gives some information about the sequence of states [7].

There are three problems that can be solved on a system that uses HMM [8]:

1. Given the observation sequence $O = O_1, O_2, \dots, O_n$ and the model $\lambda = (A, B)$, how to efficiently compute $P(O|\lambda)$, the probability of an observation sequence O . This problem can be solved by the forward algorithm.
2. Given the observation sequence $O = O_1, O_2, \dots, O_n$ and the model $\lambda = (A, B)$, how to determine the most optimal state sequence Q .

This problem can be seen in how to find the hidden parts of the Hidden Markov Models. This problem can be solved by backward algorithm.

- How to find the model $\lambda = (A, B)$ which is able to maximize $P(O|\lambda)$. This problem can be seen in how to train the model to fit the observed data. This problem can be solved by the forward-backward algorithm.

III. Research Methodology

A. Application Design and Development

GAIA application is designed and built using Java programming language in accordance with the programming language for developing applications on the Android operating system. The application design and development is divided into three steps. In the first step, we make a flow chart about the process of application running start from the beginning to the end of the process, as seen in Figure 1. The second step is to conduct word training with HMM. The third step is to design the sketch of GAIA application user interface that is easy to use and understandable by the user.

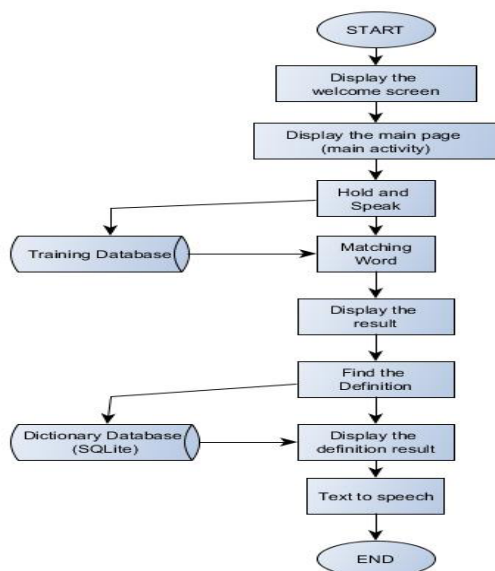


Figure 1. Flow chart of GAIA application

When firstly opened, GAIA application

shows welcome screen in a few seconds, and then switches to main activity window. In that window, a user can input his or her spoken word into an Android based smartphone by pushing and holding the ‘Hold and Speak’ (Tekan dan Bicara) button. The application will match the spoken word with one word from a set of words in the developed training database. The matched word in the database with the spoken word will appear in a text field. The definition of the word is sought in a geography dictionary, which consists of a set of geographic words or terms along with their definitions. This geography dictionary is developed in a SQLite database. The definition of the recognized word will also be shown in a text field and these texts will be further converted into speech. Thus, the user not only can read the definition of recognized word in form of texts, but also can hear the definition in form of speech. The flowchart diagram of GAIA application can be seen in Figure 1.

B. Database Training

In order to develop training database, training of series of phonemes is performed using Hidden Markov Model (HMM) approach. The goal of training is to build a set of states that represent respective set of phonemes. These states are compared with states that are built from test word in the speech recognition process. There are three steps involved in the training of database using HMM, i.e. data collection, states training, and states storing. These steps are explain in the following subsections.

B.1. Data Collection

In data collection process, we make a list of words, that is a list of geographic terms; and each word is then chopped into phonemes. These phonemes are included in a phonetic list. We use the phonetic list as in [3], where the phonemes are English phonemes that could be adopted as Indonesian phonemes. A list of sentences is made in a text file. These sentences include all words that will be trained. Each sentence line consists of ten words that could be repeated. The list of sentences is processed using Sphinx Knowledge Base Tool to produce language model for training.

The sentences according to the list made are recorded using Audacity, which results in mono audio file with .wav format, 16 bit, and 8000 Hz sampling frequency.

B.2. States Training

Training process starts by checking each word and each phoneme that is listed in the word list and phonetic list. The process cannot be continued if a mismatch between both lists is found. The next step is the beginning of training using HMM. This process trains context independent (CI) model for phonemes listed in the phonetic list and is continued with training of context dependent (CD) model for phonemes in an untied condition. The result of this training is CD untied model which is used to make a decision tree.

The next step is to make decision tree for each state from each phoneme. Then tree pruning is performed to prune unrequired states from the decision tree. The last step in HMM training is to train final model, that is CD tied model. This final model is trained in multiple stages, those are first with one Gaussian per state HMM, second with two Gaussian per state HMM and ended with three Gaussian per state HMM. This process is closed with deleting interpolation and decoding. Figure 2 summarizes the flow chart of states training.

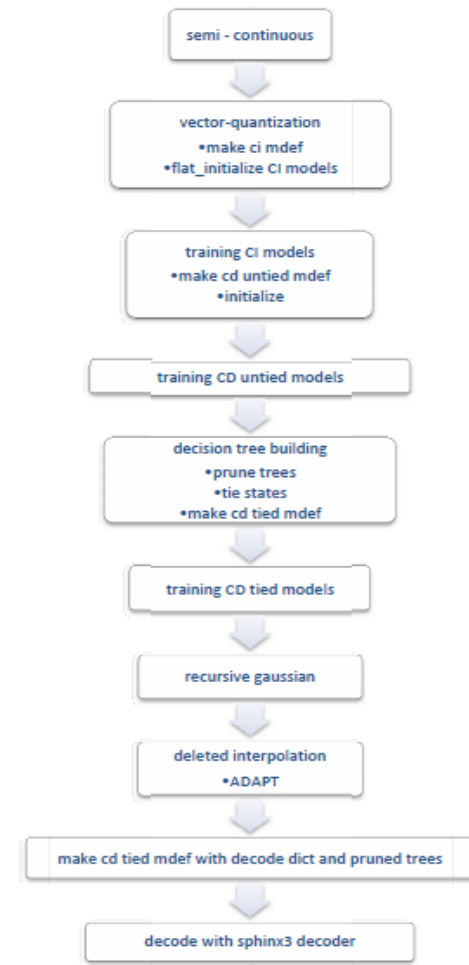


Figure 2. Flow chart of states training [6]

B.3. States Storing

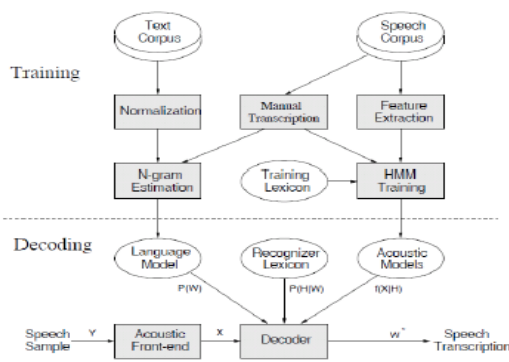
The training process results in eight files, those are mdef, feat.params, mixture_weights, means, noisedict, transition_matrices, variances, and senddump. These files together with words list and language model will further inputted to a speech recognition system as seen in Figure 3. The recognized word will also be shown in text format.

C. Application Testing

Once the application is completed, we perform testing to determine whether the application runs as expected. The application testing is based on the accuracy in matching spoken words by the user and words in the database.

D. Documentation

Documentation is done by writing the information on GAIA application program, a description of functions and variables used. Documentation is necessary for the application that has been designed so that it can be easily understood by other researchers who want to continue the research.



Gambar 3.11 Diagram proses *training* dan *decoding*
 Sumber : (Chou, Wu., Bing H.Juang, 2003, p 166).

Figure 3. Speech Recognition system [9]

IV. Implementation and Testing

A. Implementation on the Android Smartphone

Welcome screen displays the logo of the GAIA application as seen in Figure 4. When the welcome screen is running, it appears popping sound “Welcome to the GAIA application” in Indonesian. After that, the application enter the main activity.

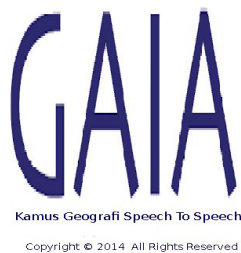


Figure 4. Welcome screen of GAIA application

In the main activity, there are five buttons, namely “Tahan dan Bicara” (Hold and Speak), “Cari Definisi” (Definition Search), “Teks ke Suara” (Text to Speech), “Tentang Kami” (About Us), and “Panduan Penggunaan” (User Manual).

In addition, there are two text fields, the text field 1 which is on the top of the page accommodates the result of the word detected and text field 2 accommodates the definition search result of the word. The display of the main activity can be seen in Figure 5 below.

The “Tahan dan Bicara” button is used to start the recording process and matching spoken word. When this button is pressed long, the application records the word spoken by the user and displays the result in the text field 1. The “Cari Definisi” button is used for the process of searching the definition of the word spoken by user, where the words of geography and their definitions have been stored in a SQLite database in advance. When the button is pressed long, the application will search into the database definition of the word that is displayed in the text field 1 (word detected from the user utterance). After that, the application displays the definition of the word in the text field 2. The “Teks ke Suara” button functions to make the process of changing the text in definition text field (text field 2) into a speech that can be heard by the user so that the definition can be learned through sound. The “Tentang Kami” button functions to describe the GAIA application briefly. The “Panduan Penggunaan” button is used to describe the guideline for using the application of GAIA.

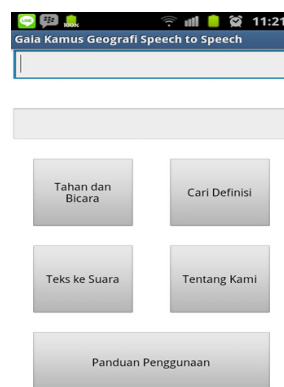


Figure 5. Main activity of GAIA application

B. Testing Result

The testing is focused on the accuracy of the prediction results when detecting spoken from a user. The testing used 50 words or geography terms in Indonesian language and ten times

testing are done for each word.

There are four conditions for the testing processes. First condition: The phone is near to the user (about 1 cm) and the environment is silent (low noise level). Second condition: The phone is near to the user (about 1 cm) and the environment is noisy (high noise level: people talking, music turn on). Third condition: The phone is far to the user (about 50 cm) and the environment is silent (low noise level). Fourth condition: The phone is far to the user (about 50 cm) and the environment is noisy (high noise level: people talking, music turn on).

The testing result in the first condition has accuracy results of 68.8%, 13%, and 76.8%. Therefore the accuracy of word recognition in the near silent condition reaches an average of 52.87%. The testing result in the second condition has accuracy results of 23.6% and 5.4%. Therefore the accuracy of word recognition in the near noisy condition reaches an average of 14.5%. Testing in the third condition has accuracy results of 36.6% and 9.8%. Therefore the accuracy of word recognition in the far silent condition reaches an average of 23.2%. Testing in fourth condition results in accuracy results of 4.2% and 1.4%. Therefore the accuracy of word recognition in the far noisy condition reaches an average of 2.8%.

The summary of testing results can be viewed in the Table 1 below [10]. From the results, GAIA application works best in the first condition (near and silent). Errors in detections can be caused by the quality of smartphone's microphone that is less sensitive.

Table 1. Testing Results of GAIA Application

Testing Condition	Accuracy in Average
Near and silent	52.87%
Near and noisy	14.5%
Far and silent	23.2%
Far and noisy	2.8%

V. Conclusion

Based on the results of research conducted, it can be concluded that the GAIA application

that designed with Pocketsphinx library using HMM approach can be constructed as an Android based speech recognition system for geography dictionary. By using the Indonesian phonemes rule, results in the near silent condition accuracy of word recognition reaches an average of 52.87%, in the near noisy reaches an average of 14.5%, in the far silent condition reaches an average of 23.2%, and in the far noisy condition reaches an average of 2.8%.

In the future works, we need to increase the number of words database. The recording of speech must be done in the soundproof room so as no noise is also recorded. One must use a good quality microphone so that the sound can be recorded clearly. The next development can use the other algorithms or approaches for speech recognition. The future research may also use other hardware media such as a tablet or smartphone with a different base operating system.

REFERENCES

- [1] B. R. Adiyanto, "Aplikasi Android Kamus Geografi Versi Bahasa Indonesia", Presentasi Karya Ilmiah Komputer, Depok: Universitas Gunadarma, Desember 2012.
- [2] T. Wijaya, S. Susanto, dan A. G. Salman, *Speech Recognition Bahasa Indonesia untuk Android*, Skripsi, Jakarta: Universitas Bina Nusantara, 2012.
- [3] V. Ferdiansyah dan A. Purwarianti, "Indonesian automatic speech recognition system using English-based acoustic model," Am. Jour. of Sig. Proc., vol. 2, no. 4, 2012, pp. 60-63, 2012.
- [4] Android developers, "Tools Overview", <http://developer.android.com/tools/help/index.html>. [accessed: 17 September 2014].
- [5] L. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*, NJ: Prentice Hall, 1993.
- [6] Carnegie Mellon University. (2011). *Building Pocketsphinx On Android* [Daring]. Avail.: <http://cmusphinx.sourceforge.net/2011/05/buildingpocketsphinx-on-android>. [accessed: 22 September 2014].
- [7] L. Rabiner, "A tutorial on hidden markov models

and selected applications in speech recognition”, Proceedings of The IEEE, vol. 77, no. 2, Februari 1989.

- [8] L. Rabiner and B. H. Juang, “Hidden Markov Models for Speech Recognition”, *Technometrics*, vol. 33, no. 3, pp. 251-272, August 1991.
- [9] W. Chou and B. H. Juang, *Pattern Recognition in Speech and Language Processing*, CRC Press. 2003.
- [10] E. Hansel, *GAlA: Rancang Bangun Aplikasi Speech Recognition pada Kamus Geografi Berbasis Android*, Skripsi, Tangerang: Universitas Multimedia Nusantara, February 2015.