Translation of English to Maguindanaon in Android.

By

Jennifer Perkeen A. Li

Colleene May Marie M. Escarlan

SCHOOL OF ARTS AND SCIENCES

ATENEO DE DAVAO UNIVERSITY

OCTOBER 2015

Translation of English to Maguindanaon in Android.

An Independent Study

Presented to

The Faculty of the Computer Studies Cluster

Ateneo de Davao University

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Information Technology

By

Jennifer Perkeen A. Li

Colleene May Marie M. Escarlan

SCHOOL OF ARTS AND SCIENCES

ATENEO DE DAVAO UNIVERSITY

OCTOBER 2015

ACKNOWLEDGMENTS

This research project would not have been possible without the help, guidance and support of several individuals who whole-heartedly extended their helping hand in the preparation and completion of this study.

First of all, we would like to thank our beloved adviser Maam Michelle Banawan who shared to us her knowledge of the subject that gave us ideas and inputs to successfully produce the final output.

To our panelists, Mr. Antonio Bulao II and Mr. Chris Alviola who helped us to make our project successful in a way of pointing out the flaws of our study to be further enhanced in the future and we would like to thank them for recognizing our efforts to complete this study.

To those who helped us for making this very project possible by helping us with the resources that we need to be able to do this project. We are very thankful to Ms. Hana Kamid, Ms. Niza Paglas, Mr. Jeffrey Ahmed Sampulna and Mr. Abdullah Adil.

To our parents, we thank them for being so supportive always by giving us their utmost and unending moral and financial support.

To our IT4A classmates, who always gave us encouragement whenever we feel like giving up and helping us in the best way that they can.

To Mr. Peter Andrei Abellana, for being our inspiration to work hard in completing this project.

To the Ateneo de Davao University community, especially to the Computer Studies Faculty, for the knowledge we have gained through the years because of their dedication in teaching us all the things we need to know that made this project possible.

And Lastly but not the least, to God Almighty, for continuously giving us strength to work on this project and for the people aforementioned above who have been around us to grace us with their presence and support. To God be the Glory!

Translation of English to Maguindanaon in Android.

COLLEENE MAY MARIE M ESCARLAN, Ateneo de Davao University
JENNIFER PERKEEN A. LI, Ateneo de Davao University

ABSTRACT

In this paper, the proponents proposes translation of English to Maguindanaon dialect in Android platform. There will be four steps in this project. Voice input is acquired through a built-in microphone then it will be process to the Android Speech Recognizer after the word or phrase is identified, it will match word or phrase then lastly it will translate the speech.

General Terms: Terms Automatic Speech Recognition (ASR), Android, Speech to Text, Machine Translation

Additional Key Words and Phrases: Speech Recognition, Feature Extraction, Fast Fourier Transform (FFT)

1. INTRODUCTION

1.1 Background of the Study

Maguindanaon is among the indigenous spoken dialect in Mindanao. Many settlers in Mindanao doesn't understand it thus, in result creating a communication gap between a Maguindanaon and a non-Maguindanaon speaker. It is also challenging to learn it because a person learns it through observation and socialization which would take time to master. This would hinder speakers of two different dialects to communicate, as a consequence, both of them cannot express properly the words they want to express.

In today's modern age, speech translation has made communication easier by translating the spoken words or phrases into another language. This is through accepting voice recording as input and translating it into desired language. Mobile speech translation is also challenging to implement as processing time and database capacity limits it.

1.2 Problem Statement

The main problem of this study is the accuracy of speech recognition. Translation using speech recognition is a very complex problem because performance of speech recognition is evaluated by its accuracy.

The specific problems of the study are as follows:

(1) Will the application provide the correct pronunciation of translated text?

(2) How can we implement speech recognition with speech matching from a database in Android?

(3) How can we implement the Maguindanaon words database in Android?

1.3 Objectives

The main objective of this study is to develop a mobile application that will recognize speech, translate it and pronounce it properly.

The specific objectives of the study are as follows:

(1) To provide the correct pronunciation of the translated text.

(2) To implement speech recognition in Android.

(3) To implement Maguindanaon words database in Android.

1.4 Significance of the Study

This study would help a non-Maguindanaon person have a conversation with a Maguindanaon person because they can understand each other. This study would also give the user a Maguindanaon equivalent of an English word that would aid the Maguindanaon speaker have an idea what the user is trying to mean. This would provide a practical benefit to those who wants to learn an ethnic dialect because one does not need to hire a personal translator which would be quite expensive. This study would also allow the user to bring the translator anywhere he goes because it is a portable translator that can be installed in smartphones with an android-based platform.

1.5 Scope and Limitations

This study's scope is translation of English language to Maguindanaon dialect with correct pronunciation of the translated text. The input of the application will be English words or phrases through speech and then, the output will Maguindanaon words or phrases in the form of text and it also includes the correct pronunciation of the words or phrases. This study will focus on smartphones with android-based platform. This study will only cover words and phrases which are conversational that are commonly used for daily communication due to limited processing power of smartphones. The translation is only one-way. The type of speech recognition to be develop is a speaker-independent which is designed to operate for any kind of speaker.

While the study's limitation is that it doesn't work without an internet connection and the translation speed depends on the speed of the internet connection. Also our database contains a limited amount of words. Another limitation is that it doesn't work well in a noisy environment. In a noisy environment the result of the user's input through voice is affected.

2. REVIEW OR RELATED LITERATURE

2.1 Translation Algorithm English to Kankanaey

In this Paper, the proponents discuss about the native dialect translated to English. They discuss Kankanaey is one of the indigenous languages and Kankanaey is widely used by people

from the Cordillera specifically from the Mountain and Benguet Provinces. Learning the language is difficult since there are no published books intended for learning. The proponents presents the theories and concepts that were adopted in the design and implementation of the translation algorithm for English to Kankanaey language using the statistical machine translator and dictionary based algorithm. It also discuss the advantages and disadvantages of the developed translation algorithm to other existing translation algorithms in terms of English to Kankanaey.

2.2 Speech to text Conversion using Android Platform

In this study, the proponents developed an application called Voice SMS that allows a user to record and convert spoken messages into a SMS text message. The Speech Recognition is done via the Internet wherein the words are retrieve from the Google's server. In their system acquires speech at run time through a microphone and processes the sampled speech to recognize the uttered text. Proponents developed this on android platform using Eclipse workbench.

2.3 Implementing SRI's Pashto Speech to Speech translation system on a smart phone

In this study, the proponents are implementing SRI's UMPC (Ultra mobile PC platform)-based Pashto speech-to-speech (S2S) translation system on a smart phone running the Android operating system. It is also stated in this study that Android has less than system memory and significantly

low processing speed with 15% loss of system accuracy. This study's user interface for the smart phone displays the speech recognition result, translation and optional backtranslation.

3. METHODOLOGY

3.1 EVALUATION

The process of evaluating the system is based on the result of the user testing to be conducted. There will be several questions to be formulated to be answered by the chosen people to test the system. The formula for the computation of results would be:

- **Mean rating of the questions answered per user**

  Formula of this is:
  N = number of questions
  X = score/rating per question
  x = Mean rating

  $x = \sum X / N$

- **Mean rating of the questions per criteria**

  N = number of question per criteria
  X = score/rating per question
  x = Mean rating

  $x = \sum X / N$

3.2 Accuracy

3.2.1 Confusion Matrix

A confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known.

Basic terms of the confusion matrix:

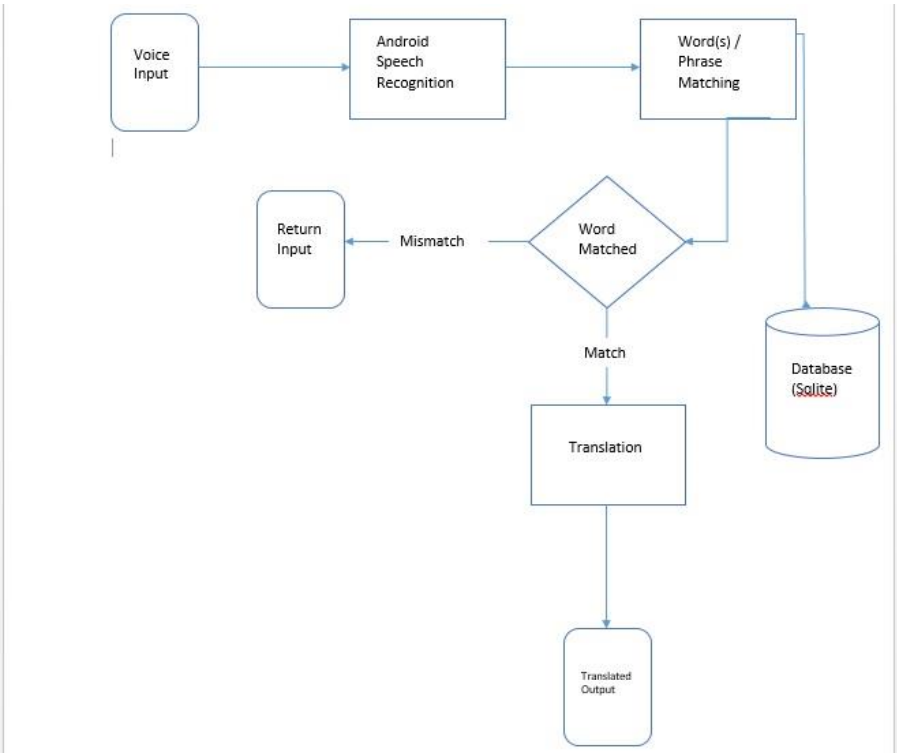
- **True positives(TP)** - These are cases in which we predicted yes and the results are yes/true.
- **True negatives(TN)** - These are cases in which we predicted no and the results are no/false.
- **False positives(FP)** - These are cases in which we predicted yes and the result showed a no/false. (Also known as a "Type I error.")
- **False negatives(FN)** - These are cases in which we predicted no and the results showed a yes/true. (Also known as a "Type II error.")

This is a list of rates that are often computed a confusion matrix:

- **Accuracy**: Overall, how often is it correct?
  - (TP + TN) / total * 100 = __ %
- **Misclassification Rate**: Overall, how often is it wrong?
  - (FP + FN) / total * 100 = __%
  - equivalent to 1 minus Accuracy
  - also known as "Error Rate"

3.3 Conceptual Framework

1.) The user would input the desired phrase or word to be translated via voice input.
2.) The *RecognizerIntent* would capture the input and send it to the Google server.
3.) After, the *RecognizerIntent* would fetch the result from the Google server.
4.) Then, the *RecognizerIntent* would return it to the same activity.
5.) Check if the word/s or phrase/s exists in the dictionary.
6.) If it exists then it would translate.
7.) If the word/s or phrase/s doesn't exists then it would return those word/s or phrase/s.



3.4 User Interface

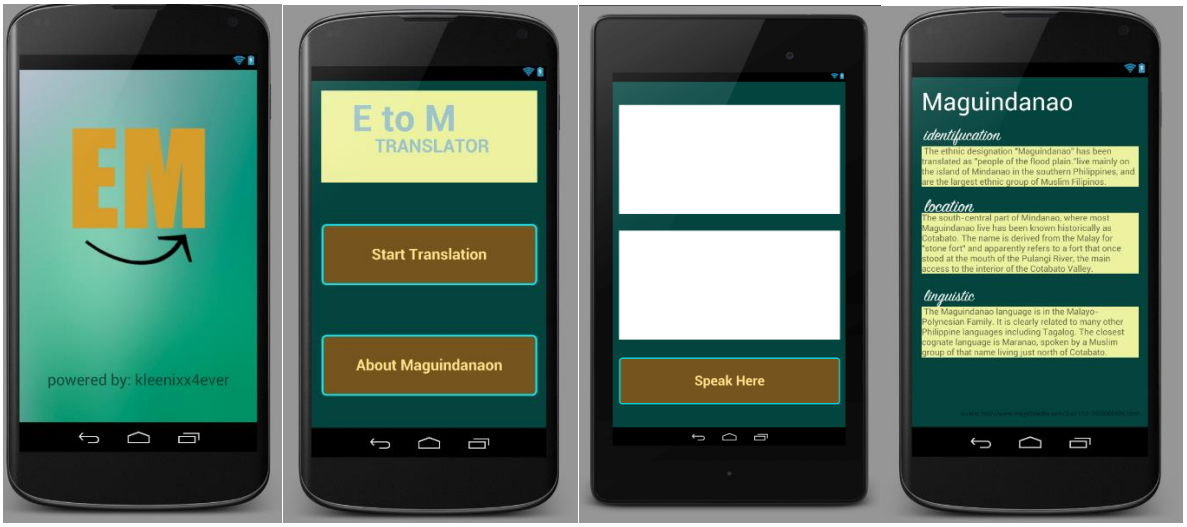


Figure 1- user interface

3.5 Database

The proponents used SQLite in building the database. SQLite is lightweight, embedded server-less database in Android and a powerful and lightweight relational database engine available to all applications. SQLite is also prepackaged on Android and does not to require to database installation or administration. SQLite supports standard relational database features like SQL syntax, transactions and prepared statements. SQLite is embedded into every Android

device. Using SQLite database in Android does not require a setup procedure or administration of the database. You only have to define the SQL statements for creating and updating the database. Afterwards the database is automatically managed for you by the Android platform. In the proponent's thesis project they have set an id which is auto-increment then in every id there is an English word or phrase and its Maguindanaon word or phrase equivalent and the URI which contains the Maguindanaon word or phrase audio file that would play after each process of the translator is done. In building the database of the translator program, the proponent's used English words and phrases with an equivalent Maguindanao words and phrases and pre-recorded audio file of Maguindanaon words and phrases they got from their sources.

	id	englishWord	mgWord	audioAddr
	Filter	Filter	Filter	Filter
1	1	one	isa	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/isaone
2	2	two	dua	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/dua_two
3	3	three	telu	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/telu_three
4	4	four	pat	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/pat_four
5	5	five	lima	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/lima_five
6	6	six	nem	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/nem_six
7	7	seven	pitu	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/pitu_seven
8	8	eight	walu	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/walu_eight
9	9	nine	siyaw	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/siyaw_nine
10	10	ten	sapulu	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/sapulu_ten
11	11	fifteen	sapulu endu lima	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/sapuluendulima_fifteen
12	12	twenty	duapulu	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/duapulu_twenty
13	13	thirty	telupulu	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/telupulu_thirty
14	14	forty	patpulu	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/patpulu_forty
15	15	fifty	limapulu	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/limapulu_fifty

	id	englishWord	mgWord	audioAddr
	Filter	Filter	Filter	Filter
242	246	sweetheart	tayan	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/isaone
243	247	i'm fine	mapiya ako bun	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/mapiya_ako_bun_imfine
244	248	how are you	ngiy betad nengka	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/ngiy_betad_nengka_ho…
245	249	i like you	galingyan ko seka	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/galingyan_ko_seka_ilike…
246	250	you are beautiful	manisan ka	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/manisan_ka_youarebea…
247	251	good morning	mapiya mapita	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/mapiya_mapita_goodmo…
248	252	good afternoon	mapiya malulem	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/mapiya_malulem_gooda…
249	253	good evening	mapiya magabi	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/mapiya_magabi_goodev…
250	254	my name is	nyako ngalan na si	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/nyako_ngalan_na_si_m…
251	255	me	saki	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/saki_im_me
252	256	i'm	saki	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/saki_im_me
253	257	mine	laki	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/laki_mine
254	258	yours	leka	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/leka_yours_your
255	259	your	nengka	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/nengka_ina
256	260	i am	saki si	android.resource://com.example.jperkeenli.maguindanaon_translator/raw/saki_si_iam

4. THEORETICAL BACKGROUND

4.1 Audio Processing

4.1.1 What is speech processing?

Auditory Processing is a natural process of taking in sound through the ear and having it travel to the language area of the brain to be interpreted. This deficit is present despite having normal hearing.

4.1.2 What is automatic speech recognition?

Automatic speech recognition (ASR) can be defined as the independent, computer-driven transcription of spoken language into readable text in real time (Stuckless, 1994). ASR is technology that allows a computer to identify the words that a person speaks into a microphone or telephone and convert it to written text.

4.1.2.1 Android API (RecognizerIntent)

Constants for supporting speech recognition through an Intent. An Intent is most significantly used in the launching of activities for an Android application. Constants that belongs to the class RecognizerIntent that we have used in our Android application are:

- **ACTION_RECOGNIZE_SPEECH –** Starts an activity that will prompt the user for speech and it sends it through a speech recognizer

- **LANGUAGE_MODEL_FREE_FORM** - Use a language model based on free-form speech recognition. A free-form speech uses a built-in very large grammar, provided by the recognizer, for a specific language.
- **EXTRA_LANGUAGE_MODEL** - Informs the recognizer which speech model to prefer when performing ACTION_RECOGNIZE_SPEECH. It is adapted to input messages in English.
- **EXTRA_LANGUAGE** - Optional IETF language tag (as defined by BCP 47), for example “en-US”.

Code Snippets:

```
private void getSpeechInput() {
    Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
    //takes the speech input then returns it d activity.

    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
    //Use a language model based on free-form speech recognition.
    // This is a value to use for EXTRA_LANGUAGE_MODEL.

    //input in free form english
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, Locale.getDefault());

    try
    {
        startActivityForResult(intent, REQ_CODE_SPEECH_INPUT);
    }

    catch (ActivityNotFoundException a)
```

figure 2 – code to create the intent to start Speech Recognizer activity

```
protected void onActivityResult(int requestCode, int resultCode, Intent data)
{
    super.onActivityResult(requestCode, resultCode, data);
    Bundle bundle = new Bundle();

    switch (requestCode)
    {
        case REQ_CODE_SPEECH_INPUT:
        {
            if (resultCode == RESULT_OK && null != data)
            {

                ArrayList<String> result = data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
                txtSpeechInput.setText(result.get(0)); //mu display ug result sa imong gi ingon
                boolean specialWord = false;
```

figure 3 – code to process the intent and extract the speech text from the intent.

4.2 Android Development

4.2.1 Android

Android is the name of the mobile operating system made by Google. It most commonly comes installed on a variety of smartphones and tablets from a host of manufacturers. Mobile application development of the operating system is based on the Java language and relies on using the software stack provided in Google android SDK (S.Holla & Katti, 2012).

4.2.2 SQLite

SQLite is a lightweight embedded server-less database in Android (W.-H, Nam, Park, & Won) and a powerful and lightweight relational database engine available to all applications. SQLite is also prepackaged on Android and does not to require to database installation or administration (Lee, 2012). SQLite supports standard relational database features like SQL syntax, transactions and prepared statements.

4.2.2.1 SQLite in Android

SQLite is embedded into every Android device. Using an SQLite database in Android does not require a setup procedure or administration of the database. You only have to define the SQL statements for creating and updating the database. Afterwards the database is automatically managed for you by the Android platform.

4.2.3 Android Studio for Android Development

Android Studio is the official integrated development environment (IDE) for Android development based on IntelliJ IDEA. It is the replacement for Eclipse.

5. RESULTS

5.1 Overall application

Respondent 1:
N = 12
X = 5 + 5 + 4 + 5 + 4 + 4 + 3 + 4 + 4 + 4 + 4 + 5
x = ΣX / N
x = 51 / 12
x = 4.25
Respondent 2:
N = 12
X = 4 + 3 + 4 + 5 + 5 + 4 + 3 + 4 + 3 + 4 + 4 +4
x = ΣX / N
x = 3.92

Respondent 3:
N = 12
X = 4 + 3 + 5 + 5 + 5 + 5 + 5 + 5 + 5 + 5 + 5 + 5
x = ΣX / N
x = 57/12
x = 4.75

Respondent 4:
N = 12

X = 4 + 3 + 5 + 5 + 5 + 5 + 5 + 5 + 5 + 5 + 5 + 5
x = ΣX / N
x = 57/12
x = 4.75

Respondent 5:
N = 12
X = 4 + 3 + 5 + 5 + 5 + 4 + 3 + 5 + 4 + 4 + 5 + 5
x = ΣX / N
x = 52/12
x = 4.33

Respondent 6:
N = 12
X = 4 + 4 + 5 + 5 + 5 + 4 + 4 + 5 + 4 + 4 + 5 + 5
x = ΣX / N
x = 54/12
x = 4.5

Respondent 7:
N = 12
X = 4 + 4+ 4 + 3 + 4 + 4 + 4 + 4+ 3 + 4 + 4+ 4
x = ΣX / N
x = 46/12
x = 3.83

	Criteria 1	**Criteria 2**	**Criteria 3**	**Criteria 4**
Respondent 1	**14**	**13**	**15**	**9**

Respondent 2	**11**	**14**	**14**	**8**
Respondent 3	**12**	**15**	**20**	**10**
Respondent 4	**12**	**15**	**20**	**10**
Respondent 5	**12**	**14**	**16**	**10**
Respondent 6	**13**	**14**	**17**	**10**
Respondent 7	**12**	**11**	**15**	**8**
Σ =	**86**	**96**	**117**	**65**

Criteria 1 : Appearance
N = 3
X = 86
x = 86/3
x = 28.67

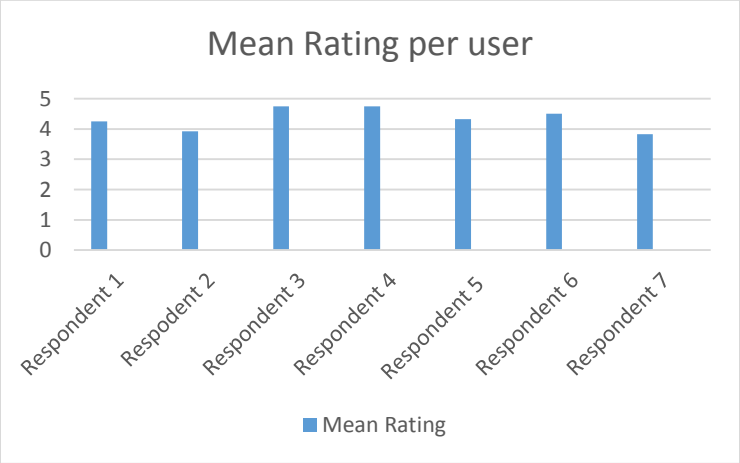
Criteria 2 : Features
N = 3
X = 96
x = 96/3
x = 32

Criteria 3 : Accuracy
N = 4
X = 117

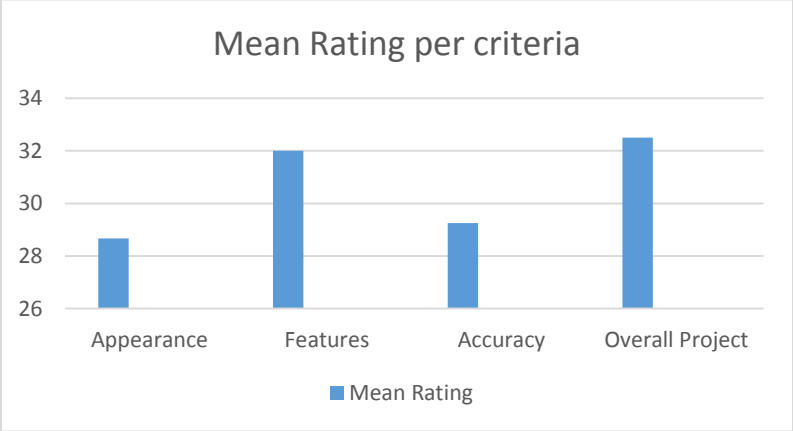
x = 117/4
x = 29.25

Criteria 4 : Features
N = 2
X = 65
x = 65/2
x = 32.5

- **Mean rating of the questions answered per user**



- **Mean rating of the questions per criteria**



5.2 Words and Phrases

- **Accuracy**:

  (TP + TN) / total * 100 = __ %
  TP – 83
  TN – 33
  total – 139
  = (83 + 33) / 139 * 100
  = 116 / 139 * 100
  = 0.83 * 100
  = 83%

- **Misclassification Rate**:

  (FP + FN) / total * 100 = __%
  FP – 19
  FN – 4
  total – 139

= (19 + 4) / 139 * 100
= 23 / 139 * 100
= 0.17 * 100
= 17%

We observe that the accuracy of the speech recognition depends on the speaker's way of pronouncing the input word/s or phrase. In our formula of solving the accuracy and error rate we predicted what would likely be the result during user testing. Would the output be correct or wrong? And our basis for prediction is the way we observe on how the speaker say/pronounce the word/s as an input.

Words should be pronounce as clearly as possible and also the environment during speaking should be quiet because the noise could affect the quality of the result. Also the stability of the internet connection could affect the translation process time.

6. CONCLUSION AND RECOMMENDATION

After a series of test, we concluded that translation with speech as an input has a lot of requirements to be fulfilled which includes speaking clearly as possible, quiet environment, and stable internet connection for a swift translation. Though the usual way of how the user speaks is deviated in return it would guarantee a higher percentage rate of accuracy.

We also concluded that the success of this project would also depend on the capability of the device to store a lot of files.

We recommend for those who wish to pursue a study related to translation that they should implement an offline speech recognition because the application is more usable for it doesn't depend on the availability of internet connection. We suggest for them to build a larger dictionary and to developed a text-to-speech for English to Maguindanaon so that the application would only consume lesser memory space.

REFERENCES

(n.d.). Retrieved from Environmental Xprt: http://www.environmental-expert.com/services/background-noise-ambient-noise-pollution-consultants-150654

Ananayo, J. E., Cayaos, J. D., & Rosal, F. G. (2011). Translation algorithm English to Kankanaey. pp. 11-16.

Angelov, K., Bringert, B., & Ranta, A. (2014). Speech-Enabled Hybrid Multilingual Translation for Mobile Devices., (pp. 41-44). Gothenburg.

Dhingra, S. D., Nijhawan, G., & Pandit, P. (2013). Isolated speech recognition using MFCC and DTW. *International Journal of Advanced Research in ELectrical, Electronics and Instrumentation Engineering, II*(8).

Dhingra, S. D., Nijhawan, G., & Pandit, P. (2013, August). Isolated Speech Recognition using MFCC and DTW. *Internation Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2*(8), 4085-4092.

Hanumante, V., Debnath, R., Bhattacharjee, D., Tripathi, D., & Roy, S. (2014). English Text to Multilingual Speech Translator Using Android. *International Journal of Inventive Engineering and Sciences (IJIES), II*.

Ishaq, B. N., & Gawali, B. W. (2014, November). Comparative Analysis of MFCC, DTW&ANN for Arabic Speech Recognition. *International Journal of Innovative Research in Advanced Engineering (IJIRAE), 1*(11), 56-61.

Kasaei, S. M., & Kasaei, S. M. (2011, Octorber). Development an Automatic Translate Voice to Sign Language Animation Based-on MFCCand Vector Quantization Method. *International Journal of Computer and Electrical Enginerring, 3*(5), 629-633.

Kotian, S., & Khadpe, A. (2011). An Optimized Approach to Voice Translation on Mobile Phones.

Lee, S. (2012). Creating and Using Databases for Android Applications. *International Journal of Database Theory and Application, 5(2)*, 99,103.

Mundt, J. C., Snyder, P. J., Cannizzaro, M. S., Chappie, K., & Geralts, D. S. (2007). Voice acoustic measures of depression severity and treatment response collected via interactive voice response (IVR) technology. *J Neurolinguistics. Author manuscript*, 50-64.

Rajsekhar, G. A. (2008). *Real Time Speaker Recognition using MFCC and VQ.* Department of Electronics & Communication Engineering National Institute of Technology.

S.Holla, & Katti, M. M. (2012). Android Based Mobile Application Development and its Security. *International Joural of Computer Trends on Technology, 3*(3), 286.

Shally Gujral, M. T. (2014, June-July). Various Issues in Computerized Speech Recognition Systems. *International Journal of Engineering Research and General Science, 2*(4).

Suryawanshi, U. J., & Ganorkar, P. D. (2014, August). Hardware Implementation of SpeechRecognition Using MFCC and Euclidean Distance. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 3*(8), 11248-11254. Retrieved March 13, 2015, from www.ijareeie.com

Tiwari, V. (2010). MFCC and its applications in speaker recognition. *International Journal on Emerging Technologies, 1(1)*, 19-12.

Vinod, P. M. (2012, December). An Approach for Classification of Dysfluent and Fluent Speech using K-NN and SVM. *International Journal of Computer Science, Engineering and Applications(UCSEA), 2*(6), 23-32.

W.-H, K., Nam, B., Park, D., & Won, Y. (n.d.). Resolving Journaling of Journal Anomaly in Android I/O: Multi-Version B-tree with Lazy Split. *12th USENIX Conference on File and Storage Technologies*, (p. 273).