

1 Voice Recognition and Android Speech to Text, TTS, Tutorial

Speak with Android phone? No Problem, Android provides such function.

1.1 Source

On github and apk could only run on android device but not on **AVM**.

1.2 Practice

1. Voice recognition,: speak to your phone,internet-permission require;
2. TTS: input string and speak in normal mode, man's tone, or woman's tone.

1.3 Environment Info

0. Kotlin
1. buildToolsVersion: '~~26.0.2~~'
2. gradle: 3.2.0-alpha15
3. Android version: minSdkVersion 19, targetSdkVersion 25

1.4 Install (Offline) Language Pack

Swipe cell android cell device up-side-down,

⚙️ → [Language & input] → [Google voice typing] → [Offline speech recognition] → [ALL]

and install the language.

1.5 Steps of project

Create a new project

Application Name: ttsdemo3

Company Domain: com.cgu.tts

1. active the internet permission (**AndroidManifest.xml**)
2. UI design: (**activity_main.xml**)

```
<textview>Input column</textview> <button>
Record</button>
<button>NormalSound</button> <button>ManSo
und</button><button>FemaleSound</button>
```

3. Java part: Voice recognition, then TextToSpeak

1.6 UI, activity_main.xml

Use the RelativeLayout :

1. create new strings in strings.xml : `

```
<resources>
    <string name="app_name">ttsdemo3</string>
    <string name="texttospeak">Input</string>
    <string name="button1">Speak</string>
</resources>
```

2. Create 4 buttons and Edittext:

```

<?xml version="1.0" encoding="utf-8"?>
    ...
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context="tts.cgu.com.ttsdemo3.MainActivity">

        <EditText
            android:id="@+id/wordToSpeak"
            android:layout_width="260dp"
            android:layout_height="wrap_content"
            android:text="@string/textttospeak" />

        <Button
            android:id="@+id/button1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/button1"
            android:layout_toRightOf="@id/wordToSpe
ak"

            android:layout_alignParentRight="true"
            android:layout_marginRight="20dp"
            android:layout_marginTop="0dp"
            android:gravity="center" />

        <Button
            android:id="@+id/btnNormal"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@+id/wordToSpeak"
            android:text="Normal Speak" />

        <Button
            android:id="@+id/btnMan"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/wordToSpeak"
            android:layout_toRightOf="@id/btnNormal
"

            android:text="Man_Speak" />

        <Button
            "" complet here, id: btnWoman""
        />

    </RelativeLayout>
    ...

```

1.7 MainActivity.java

Load modules:

```
package tts.cgu.com.ttsdemo3

import android.support.v7.app.AppCompatActivity
import android.os.Bundle
import java.util.Locale;
import android.app.Activity;
import android.content.Intent;
import android.speech.tts.TextToSpeech;
import android.widget.Button;
import android.widget.EditText;
import android.view.View;
import android.speech.RecognizerIntent;
import android.util.Log

import kotlinx.android.synthetic.main.activity_
main.*
```

1.8 Voice Recognition with TTS (kotlin)

```
class MainActivity : AppCompatActivity(),TextTo
Speech.OnInitListener {

    private var tts: TextToSpeech? = null
    private var buttonSpeak: Button? = null
    private var recognitionButton: Button? = nu
11
    private var mBtnNormal: Button? = null
    private var mBtnMan: Button? = null
    private var mBtnWoman: Button? = null
    private var mEditText: EditText? = null

    private val voiceRecognitionRequestCode = 1
004

    override fun onCreate(savedInstanceState: B
undle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        //setUpView()
```

```

        buttonSpeak = this.button1;
        recognitionButton = this.button1;
        mEditText = this.wordToSpeak;
        mBtnNormal = this.btnNormal;
        mBtnMan = this.btnMan;
        mBtnWoman = this.btnMWoman;

        buttonSpeak!!.isEnabled = false;

        tts = TextToSpeech(this, this)

        buttonSpeak!!.setOnClickListener(recognitionButtonListener)
        mBtnNormal!!.setOnClickListener{speakingNormal()}
        mBtnMan!!.setOnClickListener{speakingMan()}
        mBtnWoman!!.setOnClickListener{speakingWoman()}

    }

    private val recognitionButtonListener = View.OnClickListener { view ->
        startVoiceRecognitionActivity()
    }

    override fun onInit(status: Int) {

        if (status == TextToSpeech.SUCCESS) {
            // set US English as language for tts
            val result = tts!!.setLanguage(Locale.US)

            if (result == TextToSpeech.LANG_MISSING_DATA || result == TextToSpeech.LANG_NOT_SUPPORTED) {
                Log.e("TTS", "The Language specified is not supported!")
            } else {
                buttonSpeak!!.isEnabled = true
            }
        } else {
            Log.e("TTS", "Initialization Failed!")
        }
    }

```

```

    }

}

private fun speakOut() {
    val text = mEditText!!.text.toString()
    tts!!.speak(text, TextToSpeech.QUEUE_FLUSH, null, "")
}

private fun speakOutNormal() {
    tts!!.setPitch(1.toFloat())
    tts!!.setSpeechRate(1.toFloat())
    val text = mEditText!!.text.toString()
    tts!!.speak(text, TextToSpeech.QUEUE_FLUSH, null, "")
}

private fun speakOutMan() {
    tts!!.setPitch(0.5.toFloat())
    tts!!.setSpeechRate(0.8.toFloat())
    val text = mEditText!!.text.toString()
    tts!!.speak(text, TextToSpeech.QUEUE_FLUSH, null, "")
}

private fun speakOutWoman() {
    ... Complete Here ...
    ... Pitch:2, SpeechRate:1.5 ...
}

public override fun onDestroy() {
    // Shutdown TTS
    if (tts != null) {
        tts!!.stop()
        tts!!.shutdown()
    }
    super.onDestroy()
}

private fun startVoiceRecognitionActivity()
{
    val intent = Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH)

    intent.putExtra(RecognizerIntent.EXTRA_PROMPT, "Please say something")
    intent.putExtra(RecognizerIntent.EXTRA_

```

```

LANGUAGE_MODEL, RecognizerIntent.LANGUAGE_MODEL
_FREE_FORM)
        intent.putExtra(RecognizerIntent.EXTRA_
MAX_RESULTS, 5)
        startActivityForResult(intent, voiceRec
ognitionRequestCode)
    }

    override fun onActivityResult(requestCode:
Int, resultCode: Int, data: Intent?) {

        if(requestCode == voiceRecognitionReque
stCode && resultCode == Activity.RESULT_OK){
            val matches = data!!.getStringArray
ListExtra(RecognizerIntent.EXTRA_RESULTS)

            // The first one should be the best
one of the results
            val text = matches.first()
            mEditText!!.setText(text)
        }
        super.onActivityResult(requestCode, res
ultCode, data)
    }
}

```

1.9 Voice Recognition with TTS (Java)

```

class MainActivity : AppCompatActivity(),TextTo
Speech.OnInitListener {

    private var tts: TextToSpeech? = null
    private var buttonSpeak: Button? = null
    private var recognitionButton: Button? = nu
11
    private var mBtnNormal: Button? = null
    private var mBtnMan: Button? = null
    private var mBtnWoman: Button? = null
    private var mEditText: EditText? = null

    private val voiceRecognitionRequestCode = 1
004

    @Override
    protected void onCreate(Bundle savedInstanceState

```

```

eState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    mEditText = (EditText) findViewById(R.id.wordToSpeak);
    mBtnNormal = (Button) findViewById(R.id.btnNormal);
    mBtnMan = (Button) findViewById(R.id.btnMan);
    mBtnWoman = (Button) findViewById(R.id.btnWoman);

    tts = new TextToSpeech(getApplicationContext(),
                                new TextToSpeech
                                .OnInitListener() {
                                    @Override
                                    public void onInit(int status) {
                                        if(status != TextToSpeech.ERROR
) {
                                            // Locals.English for India
                                            tts.setLanguage(Locale.US);
                                        }
                                    }
                                });

    mBtnNormal.setOnClickListener(new View.
OnClickListener() {
        public void onClick(View v) {
            tts.setPitch((float) 1);
            tts.setSpeechRate((float) 1);
            String textToSpeak = mEditText.
getText().toString();
            if (Build.VERSION.SDK_INT >= Bu
ild.VERSION_CODES.LOLLIPOP) {
                tts.speak(textToSpeak, TextT
oSpeech.QUEUE_FLUSH, null, null);
            } else {
                tts.speak(textToSpeak, Text
ToSpeech.QUEUE_FLUSH, null);
            }
        }
    });

    mBtnMan.setOnClickListener(new View.OnC
lickListener() {

```



```

        public void onClick(View v) {
            tts.setPitch((float) 0.5);
            tts.setSpeechRate((float) 0.8);
            String textToSpeak = mEditText.
getText().toString();
            if (Build.VERSION.SDK_INT >= Bu
ild.VERSION_CODES.LOLLIPOP) {
                tts.speak(textToSpeak, TextT
oSpeech.QUEUE_FLUSH, null, null);
            } else {
                tts.speak(textToSpeak, Text
ToSpeech.QUEUE_FLUSH, null);
            }
        }
    });
    // Tone of Woman is fast and high: pict
h is 2 and rate is 1.5
    mBtnWoman.setOnClickListener(new View.O
nClickListener() {
        public void onClick(View v) {
            "" complete here as exercise ""
        }
    });

    tts=new TextToSpeech(MainActivity.this,
new TextToSpeech.OnInitListener() {
        @Override
        public void onInit(int status) {
            if(status != TextToSpeech.ERROR
) {
                // Locale.ENGLISH for India
                tts.setLanguage(Locale.US);
            } else {
                Toast.makeText(MainActivity
.this, "Initialization Failed!",
                    Toast.LENGTH
_LONG).show();
            }
        }
    });

    Intent intent = new Intent(RecognizerIn
tent.ACTION_RECOGNIZE_SPEECH);
    intent.putExtra(RecognizerIntent.EXTRA_
LANGUAGE_MODEL,

```

```

        RecognizerIntent.LANGUA
GE_MODEL_FREE_FORM);
        // Traditional Chinese is the defaulted
language, 設定辨識語言 ( 這邊設定的是繁體中文 )
        intent.putExtra(RecognizerIntent.EXTRA_
LANGUAGE, "zh-TW");
        // initialize the content, 設定語音辨識視
窗的內容
        intent.putExtra(RecognizerIntent.EXTRA_
PROMPT, "Listening...");
        startActivityForResult(intent, 1);

    }

    //flush the memory
    @Override
    public void onDestroy() {
        if (tts != null) {
            tts.stop();
            tts.shutdown();
        }
        super.onDestroy();
    }

    @Override
    protected void onActivityResult(int request
Code, int resultCode, Intent data) {
        // Store the result of Voice Recognitio
n
        // 用來儲存最後的辨識結果
        String firstMatch;
        if (requestCode == 1 && resultCode == R
ESULT_OK) {
            // store the recognized results int
o ArrayList
            // 取出多個辨識結果並儲存在 String 的 A
rrayList 中
            ArrayList<String> result =
                data.getStringArrayListExtra(
RecognizerIntent.EXTRA_RESULTS);
            firstMatch = (String) result.get(0)
;
            mEditText.setText(firstMatch);
        } else {
            firstMatch = " could not be recogni
zed ";
        }
    }

```

```
// open dialog, 開啟對話方塊
// setTitle: 設定標題
// setMessage: 設定顯示訊息 這裡會顯示辨識
的結果
new AlertDialog.Builder(MainActivity.this)
    .setTitle(" Results ")
    .setIcon(android.R.drawable.ic_
menu_search)
    .setMessage(firstMatch.toString
())
    .setPositiveButton("OK", null).
show();
}
```

In []:

--