CS378 - Mobile Computing

Speech to Text,
Text to Speech,
Telephony
SPEECH RECOGNITION
- SPEECH TO TEXT
Speech Recognition

• android.speech package
• Simplest example - start an Intent for a result
  RecognizerIntent.ACTION_RECOGNIZE_SPEECH
• uses network
  – true on the dev phones
  – doesn't work on emulator
Speech Recognition

Try the Recognizer Intent!

Press me!

LaunchActivityForResult

Say a word or phrase then it will appear on the screen.

LaunchActivityForResult

[right to, write 2, write too, write 2, right too]

Press me!
Starting Intent

```
//setup button listener
Button startButton = (Button) findViewById(R.id.trigger);
startButton.setOnClickListener(new View.OnClickListener()
{
    public void onClick(View view)
    {
        // RecognizerIntent prompts for speech and returns text
        Intent intent =
            new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);

        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
            RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
        intent.putExtra(RecognizerIntent.EXTRA_PROMPT,
            "Say a word or phrase then it will appear on the screen.");
        startActivityForResult(intent, RECOGNIZER_EXITAMPLE);
    }
});
```
Responding to Result

• Note: list of results, ordered by confidence

```java
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {

    if (requestCode == RECOGNIZER EXAMPLE
        && resultCode == RESULT_OK) {
        // returned data is a list of matches to the speech
        ArrayList<String> result =
            data.getStringArrayListExtra(
                RecognizerIntent.EXTRA_RESULTS);

        // display on screen
        tv.setText(result.toString());
    }

    super.onActivityResult(requestCode, resultCode, data);
}
Results

[start now, starting now, start now!, start Now, started now]

[Doctor Who, doctor who, Doctor Hou, doctor Hou, Dr Who]

[charter now, starter now, charger now, starting now, started now]
Modified Version

• Show confidence scores

```java
if (requestCode == RECOGNIZER_EXAMPLE
    && resultCode == RESULT_OK) {
    // returned data is a list of matches to the speech input
    ArrayList<String> result =
        data.getStringArrayResultExtra
            (RecognizerIntent.EXTRA_RESULTS);

    //display on screen
    String confKey = RecognizerIntent.EXTRA_CONFIDENCE_SCORES;
    float[] confidenceScores;
    confidenceScores = data.getFloatArrayExtra(confKey);
    tv.setText(result.toString() + "\n" +
        Arrays.toString(confidenceScores));
```
Sample Results

SpeechRecognizer

[Doctor Who, Dr Who, dr. Who, dr.who, doctor who:] [0.7612566, 0.0, 0.0, 0.0, 0.0]

Press for Talk to Text

SpeechRecognizer

[to the sea, to the C, to the city, to the Sea, to the see] [0.9412438, 0.0, 0.0, 0.0, 0.0]

Press for Talk to Text
Sample Results

[2137, 2137, 2137, 2137, 21372]
[0.78920656, 0.0, 0.0, 0.0, 0.0]

Press for Talk to Text

[1, wonderful, wonder, one, one day]
[0.2493452, 0.0, 0.0, 0.0, 0.0, 0.0]

Press for Talk to Text
Android SpeechRecognizer

- Allows lower level control of SpeechRecognizer
- If you don't want to use the intent and built in activity
- must call methods on the main activity thread
- must have RECORD_AUDIO permission
- likely uses a service
  – don't use for continuous recognition
- Requires RecognitionListener for call backs
Android TextToSpeech Class

• Not as easy to use as SpeechToText
• ... but not too bad
• In example, edit text to enter text and button to launch speech to text object
Text to Speech UI Example

computer science

Press for Text to Speech
Text To Speech

- Create Text To Speech Object
- constructor requires a Context and a TextToSpeech.OnInitListener

```java
private TextView tv;
private EditText textToSpeechET;
private Button ttsButton;
private TextToSpeech tts;

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);

    tv = (TextView) findViewById(R.id.text_result);
    textToSpeechET = (EditText) findViewById(R.id.speechToTextET);
    ttsButton = (Button) findViewById(R.id.textToSpeechButton);
    tts = new TextToSpeech(this, ttsListener);
}
```
TextToSpeech.OnInitListener

- one method, onInit
- passed an int for status

```java
private TextToSpeech.OnInitListener ttsListener
    = new TextToSpeech.OnInitListener() {
        @Override
        public void onInit(int status) {
            // based on example from http://tinyurl.com/pbg2v4r
            if (status == TextToSpeech.SUCCESS) {
                Log.d("TTS", "text to speech initialization succeeded");
                int result = tts.setLanguage(Locale.US);
                if (result == TextToSpeech.LANG_MISSING_DATA
                    || result == TextToSpeech.LANG_NOT_SUPPORTED) {
                    Log.e("TTS", "This Language is not supported");
                } else {
                    ttsButton.setEnabled(true);
                    Log.d("TTS", "locale set to US");
                }
            } else {
                Log.e("TTS", "Initialization Failed!");
            }
        }
    }
```
onClick Method for Text to Speech

• get text from EditText
• pass to TextToSpeech object

```java
public void textToSpeech(View v) {
    String textToSay = textToSpeechET.getText().toString();
    tts.speak(textToSay, TextToSpeech.QUEUE_FLUSH, null);
}
```

• Demo
• Other locales available
  – does not translate text
Text To Speech

• methods to:
• save to a sound file
• map text to sound files other than default
• check if languages available
• set rate and pitch of speech engine
TELEPHONY
Telephony

• "The working or use of telephones"
• Remember, a lot of Android devices are smartphone
• android.telephony package in the API
<table>
<thead>
<tr>
<th>Class Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CellIdentityCdma</td>
<td>CellIdentity is to represent a unique CDMA cell</td>
</tr>
<tr>
<td>CellIdentityGsm</td>
<td>CellIdentity to represent a unique GSM cell</td>
</tr>
<tr>
<td>CellIdentityLte</td>
<td>CellIdentity is to represent a unique LTE cell</td>
</tr>
<tr>
<td>CellIdentityWcdma</td>
<td>CellIdentity to represent a unique UMTS cell</td>
</tr>
<tr>
<td>CellInfo</td>
<td>Immutable cell information from a point in time.</td>
</tr>
<tr>
<td>CellInfoCdma</td>
<td>Immutable cell information from a point in time.</td>
</tr>
<tr>
<td>CellInfoGsm</td>
<td>Immutable cell information from a point in time.</td>
</tr>
<tr>
<td>CellInfoLte</td>
<td>Immutable cell information from a point in time.</td>
</tr>
<tr>
<td>CellInfoWcdma</td>
<td>Immutable cell information from a point in time.</td>
</tr>
<tr>
<td>CellLocation</td>
<td>Abstract class that represents the location of the device.</td>
</tr>
<tr>
<td>CellSignalStrength</td>
<td>Abstract base class for cell phone signal strength related information.</td>
</tr>
<tr>
<td>CellSignalStrengthCdma</td>
<td>LTE signal strength related information.</td>
</tr>
<tr>
<td>CellSignalStrengthGsm</td>
<td>GSM signal strength related information.</td>
</tr>
<tr>
<td>CellSignalStrengthLte</td>
<td>LTE signal strength related information.</td>
</tr>
<tr>
<td>CellSignalStrengthWcdma</td>
<td>Wcdma signal strength related information.</td>
</tr>
<tr>
<td>NeighboringCellInfo</td>
<td>Represents the neighboring cell information, including Received Signal Strength and others.</td>
</tr>
<tr>
<td>PhoneNumberFormattingTextWatcher</td>
<td>Watches a TextView and if a phone number is entered will format it.</td>
</tr>
<tr>
<td>PhoneNumberUtils</td>
<td>Various utilities for dealing with phone number strings.</td>
</tr>
<tr>
<td>PhoneStateListener</td>
<td>A listener class for monitoring changes in specific telephony states on the device, and others.</td>
</tr>
<tr>
<td>ServiceState</td>
<td>Contains phone state and service related information.</td>
</tr>
<tr>
<td>SignalStrength</td>
<td>Contains phone signal strength related information.</td>
</tr>
<tr>
<td>SmsManager</td>
<td>Manages SMS operations such as sending data, text, and pdu SMS messages.</td>
</tr>
<tr>
<td>SmsMessage</td>
<td>A Short Message Service message.</td>
</tr>
<tr>
<td>SmsMessage.SubmitPdu</td>
<td>Provides access to information about the telephony services on the device.</td>
</tr>
</tbody>
</table>
android.telephony package

• classes to work with
• cellular network
• the phone itself
• SMS, short messaging service, text messages
• TelephonyManager
Making Calls

• From you app, easiest way is via Intents

```java
posted_by = "111-333-222-4";

String uri = "tel:" + posted_by.trim();
Intent intent = new Intent(Intent.ACTION_CALL);
intent.setData(Uri.parse(uri));
startActivity(intent);
```

• required permission is CALL_PHONE
• Intent.ACTION_DIAL, brings up phone app with numbered entered
Telephony Provider

• Content Provider for data related to phone operation
• tables for
  – carriers, MMS, SMS