How to get the PocketSphinxAndroidDemo work

This tutorial depends on the following requirements:

A) Windows 8 64-Bit machine (Asus Vivobook)

B) Samsung Galaxy S3 with Android 4.1.2

C)Eclipse Juno 4.2.1

I wanted to start with developing some Speech To Text apps, and found a third party library that seems good to do this. This lib is called CMUSphinx, there are some versions for different systems, but in this tutorial I only want to show You, how to set up the android demo project from the CMUSphinx website. This Demo is called PocketSphinxAndroidDemo and it shows how to use pocketsphinx on an android device. It was very hard, because the tutorial on CMUSphinx website is not usefull on all systems. I searched long time about a complete tutorial for windows 8 but could´nt find some. But any day, I get this demo app work and now, I want to share my experience with all interested developers.

For dealing with CMU Sphinx, You have to download and install the android-ndk, a toolset for writing android apps in the native way, for exampe in C or C++. But no fear, even if it is good to learn how to programm with the android-ndk, the knowledge is not necessary for this. But You have to be able to build this demo, for this You need android-ndk.

1. Download ndk windows exe from developer site:

http://developer.android.com/tools/sdk/ndk/index.html

double click to start the .exe and follow instructions.

2. Download the plug-ins for Eclipse

Install CDT from Eclipse update site http://download.eclipse.org/tools/cdt/releases/indigo

Install Android SDK + NDK Plugins from Eclipse update site https://dl-ssl.google.com/android/eclipse/

Now open Eclipse as an Admin and goto Help->install new Software and adding the links above into the first field. Check all that is not checked and start.

After everything is installed, restart eclipse. After that, set ndk-path by:

window->preferences->Android->NDK and push "Browse" button to locate the installed android-ndk folder inside your machine.

3. Download CygWin

The best link I found was the RedHat Cygwin .exe:

http://www.redhat.com/services/custom/cygwin/

download the .exe and start it. From now , it will guide You through. Select a download server and go on. Now You will see the tools to download, for CMUSphinx only, choose the Folder "Devel" and click on it until at the left is showing "install". Then start installing, this needs a little bit time.

4. Download SWIG

download SWIG 1.3 (latest 1.3 version) from

http://sourceforge.net/projects/swig/files/swigwin/

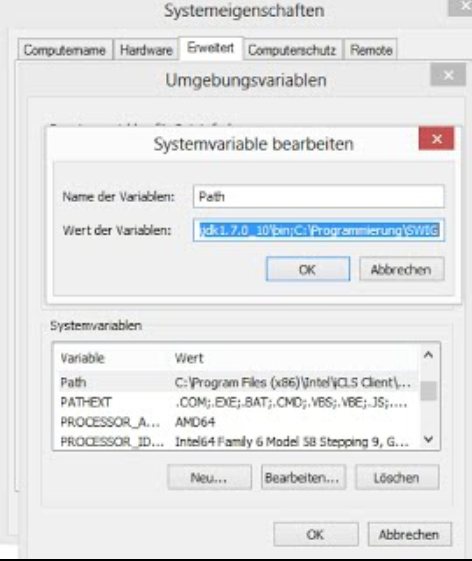
1.3 IS NEEDED BECAUSE CMUSphinx DON´T WORK WITH NEWER SWIG VERSIONS

5. Install SWIG

Installing swig on Windows is easy. But be informed that the swig.exe in the SWIG Folder , don´t work on all machines. On windows 8 I couldn´t get it work, but the only thing to do is to set the environment Variable to Your System:

At Windows go to System->System->Advanced Preferences->Environment Variables

Select System Variables and search "PATH". Select "PATH" and then press "edit". You should see something linke this:



If You have not set the Path of SWIG, You will see not SWIG like in the example above. Everything that is set before in this field: DON´T TOUCH IT. Instead, go to the end of the other declarations and set ";". Then set the path to Your SWIG Folder at Your machine, for Example

c:/Program/SWIG

That´s all for installing swig on windows.

6. Start with CMUSphinx

Download three sources from Sphinx:

sphinxbase: http://sourceforge.net/projects/cmusphinx/files/sphinxbase/0.8/

pocketsphinx: http://sourceforge.net/projects/cmusphinx/files/pocketsphinx/0.8/

PocketSphinxAndroidDemo:

cmusphinx.svn.sourceforge.net/viewvc/cmusphinx/trunk/PocketSphinxAndroidDemo/?view=tar

If any link will not work, because there is another directory meanwhile, go to the CMUSphinx homepage:

http://cmusphinx.sourceforge.net/2011/05/building-pocketsphinx-on-android/#comment-3500

BE SURE THAT YOU HAVE DOWNLOADED sphinxbase and pocketsphinx with extension .tar.gz .

DON`T DOWNLOAD THE WINZIP FILES, THEY ARE NOT FOR THIS TUTORIAL.

7. Unzip the files into one directory

Create a folder anywhere on Your machine and put all three files inside and unzip them. Now, this is very important, remove the Version declaration from the folders. The folders You have downloaded should be something like this when they are unpaked:

pocketsphinx-0.8

sphinxbase-0.8

rename them to pocketsphinx and sphinxbase, so that no version number is there. The Structure of all should look like this:

CMUSphinx (parent Folder)

|

|-pocketsphinx (subfolder of parent Folder)

|

|-sphinxbase (subfolder of parent Folder)

|

|-PocketSphinxAndroidDemo (subfolder of parent Folder)

8. Do ndk-Build in cygwin

Open the Android.mk file inside the PocketSphinxAndroidDemo Folder, for example with WordPad from windows, or Editor. Change at the Top the following line:

SPHINX\_PATH := $(HOME)/Projects/Sphinx/trunk

Set the path where Your parent CMUSphinx folder is. In windows, don´t use variables like $ to do this. Your SPHINX\_PATH should similar to this (this is my folder structure):

SPHINX\_PATH := /cygdrive/c/Programmierung/DeveloperLibs/CMUSphinx

Explanation: /cygdrive/ is called from the cygwin. There the root is called /cygdrive. The rest is the normal path where Your CMUSphinx folder is.

Press Save without changing the File extension or change the directory, only "Save".

Start the cygwin terminal as Adinistrator. If You installed Cygwin with Redhat, there are maybe two cygwin terminals. I don´t think it makes a difference, but I used the one called "Redhat Cygwin Terminal". If the terminal is initialized, do following command to go to your home.

cd c:/

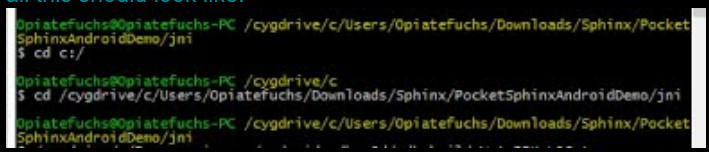
Then cd into your PocketSphinxAndroidDemo/jni folder:

cd /cygdrive/c/yourPathToTheparentCMUSphinxFolder/PocketSphinxAndroidDemo/jni

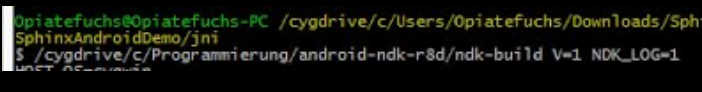
Now, do the ndk build.

/cygdrive/c/yourPathToTheNDKFolder/android-ndk-r8d/ndk-build V=1 NDK\_LOG=1

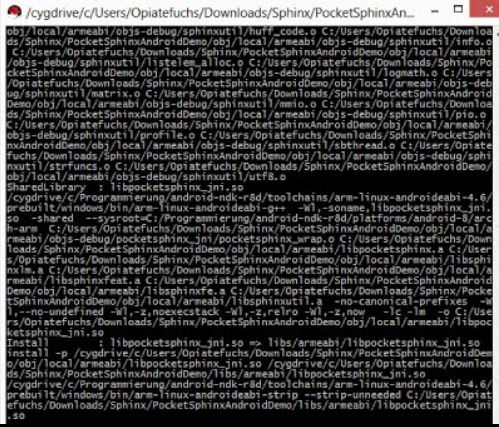
all this should look like:



and the ndk-build:



Then there must be a lot of output:



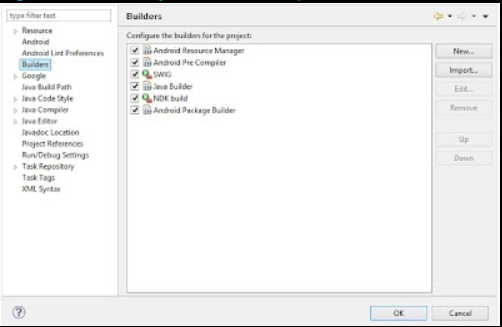
The important thing: Now check if a .so File is created in the libs Folder inside the PocketSphinxAndroidDemo/libs folder. With your windows folder-explorer, check it out and open the folder: PocketSphinxAndroidDemo->libs->armeabi-> There should be three files, gdbsetup,gdbserver and libpocketsphinx\_jni.so . If You got this, congratulation. Go to the next step.

9. Import Demo into Eclipse

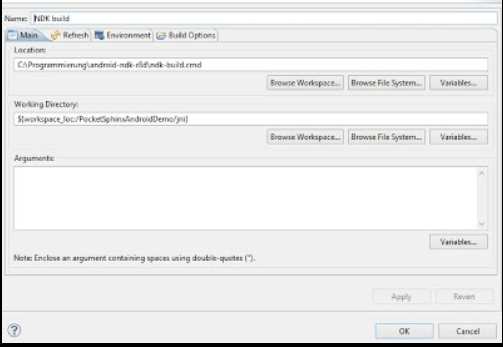
Open Eclipse, and import the Project into Your workspace. To do this, go to:

File->Import->Android->Existing Android Code into Workspace (this could be different on other Eclipse Versions). Now, select the directory from Your downloaded and unzipped PocketSphinxAndroidDemo Folder (with "Browse" Button) and push "Finish". The Project should be now in Your workspace. Note that there could be a error warning from Eclipse about double "src" folders....ignore it, this seems to be an Eclipse bug or an unnecessary warning.

Right click on this Project and select Properties->Builders. You should see something like this. Be sure that NDK and SWIG is inside:



Now, select NDK build and choose "edit". At the "Main"-Tab in "Location" , set the Path to Your ndk folder, point on the file "ndk-build" inside this folder. There are two "ndk-build" files inside, You have to point on ndk-build.cmd .



Select the Refresh Tab and choose "The project containing the selected resource". Next, go to the "Build-Options" Tab und deselect "Specify working set of relevant resources". Select Apply and Ok.

Now, select the SWIG and choose "edit" (the same way like with NDK-Build). Don´t change anything in the "Main"-Tab, If You downloaded SWIG like here at the beginning of this tutorial, You don´t have to change anything. In the refresh tab select “The folder containing the selected resource” and in the Build Options tab deselect “Specifiy working set of relevant resources”, like You did it with NDK-Build.

Press Apply and close this window. Your Eclipse Project is ready, now You have to do some changes inside Your phone´s sd card or internal storage.

10. Set up Your Phone

It´s time to set up Your phone. Create some Folders on Your sd Card. Note, that there are phones with two storage directories. One is the SD Card, the other one is the internal Storage. I created the folder at my internal storage, but maybe the data will not grabbed from the right directory. I recommend to create this following folder structure on both directories, if Your phone got both.

First, create the folder called "edu.cmu.pocketsphinx ". Inside this folder, create the following folder structure:

edu.cmu.pocketsphinx --> (created folder)

|

----> hmm --> (subfolder of edu.cmu.pocketsphinx)

| |

| ----> en\_US -->(subfolder of hmm)

| |

| ----> hub4wsj\_sc\_8k -->(subfolder of hmm)

|

----> lm-->(subfolder of edu.cmu.pocketsphinx)

|

-----> en\_US-->(subfolder of edu.cmu.pocketsphinx)

We have top copy some files from our pocketsphinx folder at our computer to the created directory on the phone:

On your computer, go to your pocketsphinx folder: pocketsphinx\model\hmm\en\_US and copy the files to the phones directory "edu.cmu.pocketsphinx/hmm/en\_US".

Next, go to your computers directory "pocketsphinx/model/lm/" and copy the files to your phones directory "edu.cmu.pocketsphinx/lm/en\_US".

Now, in Eclipse, open the RecognizerTask.java file on the src folder in package edu.cmu.pocketsphinx.demo . You have to change the path´ for getting the files we copied before. AND NOW, THIS IS VERY IMPORTANT: At the tutorial from CMU Sphinx, there is a path declared that not work on every phone. Instead of doing changes like described in the tutorial, change these path´ like this:

Search the constructor "public ReconizerTask{}", it must be somewhere at line 160 (less or more). Inside the constructor, change the path´, the constructor have to look like this:

public RecognizerTask() {

String sdPath = Environment.getExternalStorageDirectory().toString();

pocketsphinx.setLogfile(sdPath+"/edu.cmu.pocketsphinx/pocketsphinx.log");

Config c = new Config();

c.setString("-hmm",sdPath+"/edu.cmu.pocketsphinx/hmm/en\_US/hub4wsj\_sc\_8k");

c.setString("-dict",sdPath+"/edu.cmu.pocketsphinx/lm/en\_US/hub4.5000.dic");

c.setString("-lm",sdPath+"/edu.cmu.pocketsphinx/lm/en\_US/hub4.5000.DMP");

c.setString("-rawlogdir", sdPath+"/edu.cmu.pocketsphinx");

c.setFloat("-samprate", 8000.0);

c.setInt("-maxhmmpf", 2000);

c.setInt("-maxwpf", 10);

c.setInt("-pl\_window", 2);

c.setBoolean("-backtrace", true);

c.setBoolean("-bestpath", false);

this.ps = new Decoder(c);

this.audio = null;

this.audioq = new LinkedBlockingQueue<short[]>();

this.use\_partials = false;

this.mailbox = Event.NONE;

}

If You created the folder structure on both directories like I said before, You will get sure that the demo can grab the files. On my mobile phone ,there is no sd card inside, so I got only one directorie, and this works very fine.

Well, thats it, now You should be able to build and run the project. The PocketSphinxAndroidDemo is just the basic of dealing with CMUSphinx. For example, this demo only listen to english words, but it is possible to get more language models. If I go forward with this, I will write a tutorial for how to build an own PocketSphinx Application. Until then, I hope You have success with this tutorial. It´s possible, that this does not work on all windows machines, I could only do this tutorial for a windows 8 64-Bit machine, but if there are some differences to other windows versions, please let me know. If possibe, I will integrate this into this tutorial to help other poeple.

If You have detected some wrong descriptions, also let me know, so I could fix this.