VoiceBridge Getting Started Guide

VoiceBridge is an **open source** (AI-TOOLKIT Open Source License - Apache 2.0 based, very permissive and allows commercial use) **speech recognition C++ toolkit** optimized for MS Windows 64-bit (can be easily modified to compile on other operating systems). VoiceBridge fills the gap for MS Windows speech recognition developers.

VoiceBridge Official website:

AI-TOOLKIT VoiceBridge: https://ai-toolkit.blogspot.com/p/voicebridge.html

VoiceBridge can be considered to be the MS Windows counterpart of KALDI (speech recognition software for Unix like operating systems) with the following differences and extensions:

1. VoiceBridge is C++ only code without any scripts. Kaldi depends heavily on several scripting languages (Bash, Perl, and Python).

2. The **aim** of VoiceBridge is to make writing **high quality professional and fast speech recognition software** very easy. VoiceBridge does not include all of the available models in Kaldi but a selection of models which provide very good accuracy and are fast. Kaldi is a research system and will always have more models available. VoiceBridge may add new models in the future if they provide significant accuracy and/or speed improvement.

   The following models are currently available in VoiceBridge:
   a. Monophone,
   b. Delta + delta-delta triphone,
   c. LDA+MLLT,
   d. LDA+MLLT+SAT,
   e. DELTA+SAT (delta + delta-delta + SAT).

   Currently the DELTA+SAT is the best performing model with the highest accuracy and speed (1/5th of the training time compared to LDA+MLLT+SAT). Due to the automatic tuning of some input (e.g. pronunciation) VoiceBridge achieves the same accuracy in case of the clean LibriSpeech data as the DNN model in Kaldi!

3. VoiceBridge includes the following extra modules not included in Kaldi:
   a. **Automatic language model generation**.
   b. **Automatic pronunciation lexicon generation**.

   Thanks to these modules VoiceBridge only requires a limited number of input:
   a. Wav files.
   b. Text transcription files for each wav file.
   c. Reference language dictionary (available in the VoiceBridge distribution).

   Your speech recognition job has just become much easier!

4. VoiceBridge is **hardware accelerated** in two ways:
    a. Automatic parallel processing by automatic CPU/core detection and work distribution. More processors/processor cores mean faster processing!
    b. VoiceBridge makes use of the Intel Math Kernel Library (MKL) which further accelerates processing by making use of special processor instruction sets.

    Note: VoiceBridge currently does not support grid computing and CUDA and there are also no plans to add these in the near future.

5. The VoiceBridge C++ code is organized in **1 DLL library**. This is a huge difference between Kaldi and VoiceBridge because Kaldi includes hundreds of exe and script files. For this reason it is very easy to distribute your software built upon VoiceBridge. VoiceBridge is aimed to be fast, high accuracy and easy to use professional production ready system.

6. VoiceBridge includes two complete **examples** which demonstrate how to use the library. Both examples are also available in Kaldi. This makes the learning of VoiceBridge for Kaldi users much easier.

    One of the examples is the **Yes-No example**. This is a very simple speech recognition example in which we train a model to recognize people saying 'yes' or 'no'. The WER (word error rate) of this example in VoiceBridge is 2% (98% accuracy) and the training + testing takes about 8 seconds (with 4 processor cores).

    The second example is the so called **LibriSpeech example**, a real world speech recognition application in which several hours of English speech learning and recognition are included. The WER (word error rate) of this example in VoiceBridge is 5.92% (94% accuracy) and the training + testing takes about 25 minutes (with 4 processor cores).

    Both examples are ready to use code templates for your speech recognition projects! More examples may be added later.

7. Everything is included in the VoiceBridge distribution except the Intel MKL library which can be downloaded for free from this website: Intel MKL: https://software.intel.com/en-us/mkl.

8. **Compilation**: VoiceBridge compilation can be done with the included MS Visual Studio 2017 projects. As you probably know MS VS 2017 is free software. VoiceBridge only supports 64-bit compilation because 64-bit systems are faster and can use more memory.

    Please follow these steps for the compilation:
    a. Download and install the Intel MKL library. Note the location of the library. For example:

        C:\IntelSWTools\compilers_and_libraries_2018\windows\mkl

b. Adjust the MKL library location in the 'SettingsVoiceBridge.props' file in the root directory of VoiceBridge. Do not modify anything else because VoiceBridge is setup with relative paths and therefore you do not need to adjust any more settings.

c. Compile the openfst project located in 'VoiceBridge\openfst-win-1.6'. It is best to compile both Debug and Release versions. **Important**: Whole program optimization must be OFF for the library!

d. Compile the Kaldi project located in 'D:_WORK1\VoiceBridge\kaldi-master'. It is best to compile both Debug and Release versions. **Important**: Whole program optimization must be OFF for the library!

e. Compile the VoiceBridge DLL located in 'VoiceBridge\VoiceBridge\VoiceBridge'.
Important: Whole program optimization must be OFF! This option could result in 2-3% speed improvement but then the DLL should be cut in peaces because VS can not handle the optimization of so much code.

Note: Please note that there is a shortcut to all of the above mentioned VS2017 projects in the root directory of the distribution.

f. In the TestDll example you can select which example you want to run. Choose between 'TestYesNo();' or 'TestLibriSpeech();' or run both after each other.

Important: You must make sure that the path to the example projects is correct in both example cpp files (YesNo.cpp, LibriSpeech.cpp). E.g. for the Yes-No example the path is set with the following command:

fs::path project(exepath.branch_path() / "../../../../../VoiceBridgeProjects/YesNo");

Do this after downloading the example projects from the Github repository:
VoiceBridgeProjects: https://github.com/AI-TOOLKIT/VoiceBridgeProjects (the data is ~600 MB). If you put the example projects into a directory called VoiceBridgeProjects at the same level as the VoiceBridge directory (e.g.: C:\VoiceBridge and C:\ VoiceBridgeProjects) then you do not need to change anything. In this case the input directory for the Yes-No project would be located in: 'C:\VoiceBridgeProjects\YesNo\input'.

g. Compile the test project located in 'VoiceBridge\VoiceBridge\TestDll'.

h. Run the example.

Important: You must copy the dll's from the Redistributables directory into the Release and/or Debug folder!

9. **Redistribution**: The directory 'VoiceBridge\Redistributables' contains all the necessary dll's which need to be redistributed with any software built with the use of VoiceBridge. Most of them are for the Intel MKL library and one is for OpenMP

support. You may of course need to distribute some more dll's required by your compiler (MS VS2017) for example for the C++ runtime.

Note: Please note that the MKL dll's are from the w_mkl_2018.1.156 distribution. You may need to replace these if you download a more recent version later!

10. **Documentation**: In speech recognition technical matters please refer to the in the VoiceBridge distribution included e-books in PDF format and to the Kaldi documentation here: [Kaldi documentation](http://kaldi-asr.org/doc/about.html): http://kaldi-asr.org/doc/about.html. For all other subjects concerning the VoiceBridge library and options please consult this website and the heavily documented source code.

Attribution

VoiceBridge would not have been possible without the work of the following people and companies:

1. **Daniel Povey** – Dan is the main developer of Kaldi (http://kaldi-asr.org/doc/about.html) and an exceptional researcher and person. Dan was a great help during the making of VoiceBridge.

2. **Many people who contributed to Kaldi**. Please consult the Kaldi website for a full list of names.

3. **Josef Robert Novak** – Josef has developed Phonetisaurus on which the automatic VoiceBridge pronunciation generator is based on.

4. **Massachusetts Institute of Technology (MIT)** – Several people at MIT contributed to the MITLM project on which the VoiceBridge automatic language model generator is based on.

5. **Microsoft Corporation** – many of the Kaldi modules (also written by Dan while working at Microsoft) are included in VoiceBridge.

6. **Johns Hopkins University** – several people contributed to the Kaldi project.

7. **Google Inc.**

8. **Arash Partow** – Arash has developed the indispensable String Toolkit Library (http://www.partow.net/programming/strtk/index.html).

9. **Boost developers** (www.boost.org)

10. There are most probably still many people and companies who contributed to projects who are not mentioned here above but their names can be found in the source code. If you feel that you are a major contributor and I have forgot to mention your name then please let me know and I will add your name.

The use of VoiceBridge is subject to the AI-TOOLKIT Open Source Software License:

AI-TOOLKIT Open Source Software License - Version 2.0 - January 9, 2018

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the Apache 2.0 license and the following:

1. You may not remove any copyright and license information from the source code.
2. The following statement must be included in the help file and also in the splash/about box in all copies of the Software, in whole or in part, and in all derivative works of the Software, including software in the form of machine-executable object code:

"Portions Copyright Zoltan Somogyi (AI-TOOLKIT), license:
AI-TOOLKIT Open Source Software License - Version 2.0 - January 9, 2018
https://ai-toolkit.blogspot.com/p/ai-toolkit-open-source-software-license.html."

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

You may obtain a copy of the Apache 2.0 License at: http://www.apache.org/licenses/LICENSE-2.0

This document has been last modified: Saturday, 10 February, 2018